
gstswitch Documentation

Release

hyades

January 16, 2015

1	A replacement of DVSwitch based on gstreamer	1
1.1	gst-switch python API	1
1.2	Unittest for Python-API	10
1.3	Integration Tests for gstswitch and Python-API	25
2	Indices and tables	27
	Python Module Index	29

A replacement of DVSwitch based on gstreamer

Python - API:

1.1 gst-switch python API

1.1.1 gstswitch Package

Python API for gst-switch

1.1.2 connection Module

connection deals with all low level method calls over dbus The Connection object is capable of invoking remote methods over dbus

```
class gstswitch.connection.Connection (address='unix:abstract=gstswitch',
                                         bus_name='info.duzy.gst.switch.SwitchController', ob-
                                         ject_path='/info/duzy/gst/switch/SwitchController', de-
                                         fault_interface='info.duzy.gst.switch.SwitchControllerInterface')
```

Bases: object

Class which makes all remote object class. Deals with lower level connection and remote method invoking

Default bus-address unix:abstract=gstswitch

Param None

CONNECTION_FLAGS = 1

address

Get the address

adjust_pip (xpos, ypos, width, height)

adjust_pip(in i dx, in i dy, in i dw, in i dh, out u result);

Calls adjust_pip remotely

Parameters

- **xpos** – the X position of the PIP
- **ypos** – the Y position of the PIP
- **width** – the width of the PIP

- **height** – the height of the PIP

Returns tuple with first element as result -

PIP has been changed successfully

bus_name

Get the bus name

click_video (*xpos, ypos, width, height*)

click_video(in i x, in i y, in i fw, in i fh, out b result);

Calls click_video remotely

Parameters

- **xpos** –
- **ypos** –
- **width** –
- **height** –

Returns tuple with first element True if requested

connect_dbus ()

Make a new connection using the parameters belonging to the class to the gst-switch-srv over dbus. Sets the self.connection

Params None

Returns Nothing

Raises ConnectionError GError occurs while making a connection

default_interface

Get the default interface

get_audio_port ()

get_audio_port(out i port); Calls get_audio_port remotely

Param None

Returns tuple with first element audio port number

get_compose_port ()

get_compose_port(out i port); Calls get_compose_port remotely

Param None

Returns tuple with first element compose port number

get_encode_port ()

get_encode_port(out i port); Calls get_encode_port remotely

Param None

Returns tuple with first element encode port number

get_preview_ports ()

get_preview_ports(out s ports); Calls get_preview_ports remotely

Param None

Returns tuple with first element a string in the form of

'[(3002, 1, 7), (3003, 1, 8)]'

mark_face (*faces*)
 mark_face(in a(iiii) faces); Calls mark_face remotely
Parameters **faces** – tuple having four elements
Returns tuple with first element True if requested

mark_tracking (*faces*)
 mark_tracking(in a(iiii) faces); Calls mark_tracking remotely
Parameters **faces** – tuple having four elements
Returns tuple with first element True if requested

new_record ()
 new_record(out b result); Calls new_record remotely
Param None:
 returns: tuple with first element True if requested

object_path
 Get the object path

set_composite_mode (*mode*)
 set_composite_mode(in i channel, out b result);
 Calls set_composite_mode remotely
Parameters **mode** – new composite mode
Returns tuple with first element True if requested

set_encode_mode (*channel*)
 set_encode_mode(in i channel, out b result);
 Calls set_encode_mode remotely **Does not do anything**
Param channel
Returns tuple with first element True if requested

switch (*channel, port*)
 switch(in i channel, in i port, out b result);
 Calls switch remotely
Parameters

- **channel** – The channel to be switched, ‘A’, ‘B’, ‘a’
- **port** – The target port number

Returns tuple with first element True if requested

1.1.3 controller Module

The controller is the interface for all remote method calls over dbus. The Controller class creates the controller, which can be used to invoke the remote methods.

```
class gstswitch.controller.Controller (address='unix:abstract=gstswitch',  
                                       bus_name='info.duzy.gst.switch.SwitchController', ob-  
                                       ject_path='/info/duzy/gst/switch/SwitchController', de-  
                                       fault_interface='info.duzy.gst.switch.SwitchControllerInterface')
```

Bases: object

A Class to control all interactions with the gst-switch-srv over dbus. Provides the interface for higher level interactions

Param None

address

Get the address

adjust_pip (*xpos, ypos, width, height*)

Change the PIP position and size

Parameters

- **xpos** – the x position of the PIP
- **ypos** – the y position of the PIP
- **width** – the width of the PIP
- **height** – the height of the PIP

Returns result - PIP has been changed successfully

bus_name

Get the bus name

click_video (*xpos, ypos, width, height*)

User click on the video

Parameters

- **xpos** –
- **ypos** –
- **width** –
- **height** –

Returns True when requested

default_interface

Get the default interface

establish_connection ()

Establishes a fresh connection to the dbus Connection stored as self.connection

Param None

Returns None

get_audio_port ()

Get the audio port number

Param None

Returns audio port number

get_compose_port ()

Get the compose port number

Param None

Returns compose port number

get_encode_port ()

Get the encode port number

Param None

Returns encode port number

get_preview_ports ()

Get all the preview ports

Param None

Returns list of all preview ports

mark_face (*faces*)

Mark faces

Parameters *faces* – tuple having four elements

Returns True when requested

mark_tracking (*faces*)

Mark tracking

Parameters *faces* – tuple having four elements

Returns True when requested

new_record ()

Start a new recording

Param None

object_path

Get the object path

classmethod parse_preview_ports (*res*)

Parses the preview_ports string

set_composite_mode (*mode*)

Set the current composite mode. Modes between 0 and 3 are allowed.

Parameters *mode* – new composite mode

Returns True when requested

set_encode_mode (*channel*)

Set the encode mode WARNING: THIS DOES NOT WORK.

Param channel

Returns True when requested

switch (*channel, port*)

Switch the channel to the target port

Parameters

- **channel** – The channel to be switched, 'A', 'B', 'a'
- **port** – The target port number

Returns True when requested

1.1.4 exception Module

All custom exceptions come here

exception `gstswitch.exception.BaseError`

Bases: `exceptions.Exception`

docstring for BaseError

exception `gstswitch.exception.PathError`

Bases: `gstswitch.exception.BaseError`

docstring for PathError

exception `gstswitch.exception.ServerProcessError`

Bases: `gstswitch.exception.BaseError`

docstring for ServerProcessError

exception `gstswitch.exception.ConnectionError`

Bases: `gstswitch.exception.BaseError`

docstring for ConnectionError

exception `gstswitch.exception.ConnectionReturnError`

Bases: `gstswitch.exception.BaseError`

docstring for ConnectionReturnError

exception `gstswitch.exception.RangeError`

Bases: `gstswitch.exception.BaseError`

docstring for RangeError

exception `gstswitch.exception.InvalidIndexError`

Bases: `gstswitch.exception.BaseError`

docstring for InvalidIndexError

1.1.5 helpers Module

Has helper classes which create test video and audio sources. It is also possible to create a preview out source showing the compose port output.

class `gstswitch.helpers.TestSources` (*video_port=None, audio_port=None*)

Bases: `object`

A Controller of test sources feeding into the gst-switch-srv :param width: The width of the output video :param height: The height of the output video :param pattern: The videotestsrc pattern of the output video :param timeoverlay: True to enable a running time over video :param clockoverlay: True to enable current clock time over video

audio_port

Get the audio port

get_test_audio()

Returns a list of processes acting as audio test sources running :returns: A list containing all audio test sources running

get_test_video()

Returns a list of processes acting as video test sources running :returns: A list containing all video test sources running

new_test_audio (*freq=110, wave=None*)

Start a new test audio :param port: The port of where the TCP stream will be sent Should be same as audio port of gst-switch-src :param width: The width of the output audio :param height: The height of the output audio :param pattern: The audiotestsrc pattern of the output audio :param timeoverlay: True to enable a running time over audio :param clockoverlay: True to enable current clock time over audio

new_test_video (*width=300, height=200, pattern=None, timeoverlay=False, clockoverlay=False*)

Start a new test video :param port: The port of where the TCP stream will be sent Should be same as video port of gst-switch-src :param width: The width of the output video :param height: The height of the output video :param pattern: The videotestsrc pattern of the output video :param timeoverlay: True to enable a running time over video :param clockoverlay: True to enable current clock time over video

running_tests_audio

Get the currently running test audio list

running_tests_video

Get the currently running test video list

terminate_audio ()

Terminate all test audio sources

terminate_index_audio (*index*)

Terminate audio test source specified by index :param index: The index of the audio source to terminate Use `get_test_audio` for finding the index

terminate_index_video (*index*)

Terminate video test source specified by index :param index: The index of the video source to terminate Use `get_test_video` for finding the index

terminate_video ()

Terminate all test video sources

video_port

Get the video port

class `gstswitch.helpers.PreviewSinks` (*preview_port=3001*)

Bases: `object`

A Controller for preview sinks to preview ports of gst-switch-srv :param preview_port: The preview port to get the preview

preview_port

Get the preview port

run ()

Run the Preview Sink

terminate ()

End/Terminate the Preview Sink

1.1.6 server Module

The server deals with all operations controlling gst-switch-srv These include all OS related tasks

class `gstswitch.server.Server` (*path=None, video_port=3000, audio_port=4000, control_port=5000, record_file=False*)

Bases: `object`

Control all server related operations

Parameters `path` – Path where the executable gst-switch-srv

is located. Provide the full path. By default looks in the current \$PATH. :param video_port: The video port number - default = 3000 :param audio_port: The audio port number - default = 4000 :param control_port: The control port number - default = 5000 :param record_file: The record file format :returns: nothing

SLEEP_TIME = 0.5

audio_port

Get the audio port

control_port

Get the control port

gcov_flush()

Generate gcov coverage by sending the signal SIGUSR1 The generated gcda files are dumped in tools directory. Does not kill the process

Param None

Returns True when success

Raises

- **ServerProcessError** – If Server is not running
- **ServerProcessError** – Unable to send signal

kill (*cov=False*)

Kill the server process by sending signal.SIGKILL self.proc is made None on success

Param None

Returns True when success

Raises

- **ServerProcessError** – Process does not exist
- **ServerProcessError** – Cannot kill process

classmethod make_coverage()

Generate coverage Calls 'make coverage' to generate coverage in .gcov files

path

Get the path

record_file

Get the record file

run (*gst_option=''*)

Launch the server process

Param None

Gst-option Any gstreamer option.

Refer to <http://www.linuxmanpages.com/man1/gst-launch-0.8.1.php#lBAF>. Multiple can be added separated by spaces :returns: nothing :raises IOError: Fail to open /dev/null (os.devnull) :raises PathError: Unable to find gst-switch-srv at path specified :raises ServerProcessError: Running gst-switch-srv gives a OS based error.

terminate (*cov=False*)

Terminate the server. self.proc is made None on success

Param None

Returns True when success

Raises

- **ServerProcessError** – Process does not exist
- **ServerProcessError** – Cannot terminate process. Try killing it

video_port

Get the video port

1.1.7 testsource Module

The testsource contains all gstreamer pipelines It provides the abse for all the other gstreamer components are build upon.

```
class gstswitch.testsource.Preview(port)
```

Bases: object

A Preview Element :param port: The preview port

end()

End/disable the pipeline

pause()

Pause the pipeline

preview_port

Get the Preview Port

run()

Run the pipeline

```
class gstswitch.testsource.VideoSrc(port, width=300, height=200, pattern=None, timeover-
                                     lay=False, clockoverlay=False)
```

Bases: object

A Test Video Source :param width: The width of the output video :param height: The height of the output video :param pattern: The videotestsrc pattern of the output video None for random :param timeoverlay: True to enable a running time over video :param clockoverlay: True to enable current clock time over video

HOST = 'localhost'**clockoverlay**

Get the Clockoverlay

end()

End/disable the pipeline

classmethod generate_pattern(pattern)

Generate a random pattern if not specified

height

Get the height

pattern

Get the Pattern

pause()

End the pipeline

port

Get the Video Port

```
run ()
    Run the pipeline

timeoverlay
    Get the timeoverlay

width
    Get the width

class gstswitch.testsource.AudioSrc (port, freq=110, wave=None)
    Bases: object
    docstring for AudioSrc
    HOST = 'localhost'

end ()
    End/disable the pipeline

freq
    Get the Frequency

classmethod generate_wave (wave)
    Generate a random wave if not specified

pause ()
    End the pipeline

port
    Get the Audio Port

run ()
    Run the pipeline

wave
    Get the wave number
```

Tests:

1.2 Unittest for Python-API

1.2.1 unittests Package

Unittests for gst-switch Python-API

1.2.2 test_connection_unit Module

Unittests for Connection class in connection.py

```
class unittests.test_connection_unit.MockConnection (method)
    Bases: object
    A class which mocks the Connection class

call_sync (bus_name, object_path, interface_name, method_name, parameters, reply_type, flags,
            timeout_msec, cancellable)
    Mock of call_sync method, raises GLib.GError if interface_name invalid

funcs = {'mark_tracking': None, 'get_audio_port': (4000,), 'click_video': (True,), 'adjust_pip': (1), 'set_encode_mode':
```

```

class unittests.test_connection_unit.TestAddress
    Bases: object

    unittests for address parameter

    test_address_colon ()
        Test if address has no colon

    test_address_normal ()
        "Test if address is valid

    test_address_null ()
        Test if address is null

class unittests.test_connection_unit.TestBusName
    Bases: object

    unittests for bus_name parameter

    test_normal ()
        Test if bus_name is not null

    test_normal_none ()
        Test if bus_name is null

class unittests.test_connection_unit.TestConnectDBus
    Bases: object

    unittests for the connect_dbus method of Connection class

    test_bad_address ()
        Test if wrong address is given - 1

    test_bad_address2 ()
        Test if wrong address is given - 2

    test_bad_address3 ()
        Test if wrong address is given - 3

    test_mock1 (monkeypatch)
        Test GLib.GError exception

    test_mock2 (monkeypatch)
        Test GLib.GError exception

class unittests.test_connection_unit.TestInterface
    Bases: object

    unittests for default_interface parameter

    test_interface_dot ()
        Test when the default_interface has <2 dots

    test_interface_none ()
        Test if default_interface is null

    test_interface_normal ()
        Test if default_interface is valid

class unittests.test_connection_unit.TestObjectPath
    Bases: object

    unittests for object_path parameter

```

```
test_object_path_blank()
    Test if object_path is null

test_object_path_normal()
    Test of object_path is valid

test_object_path_slash()
    Test when object_path doesn't have slash in start

unittests.test_connection_unit.test_adjust_pip()
    Test the adjust_pip method

unittests.test_connection_unit.test_click_video()
    Test the click_video method

unittests.test_connection_unit.test_get_audio_port()
    Test the get_audio_port method

unittests.test_connection_unit.test_get_compose_port()
    Test the get_compose_port method

unittests.test_connection_unit.test_get_encode_port()
    Test the get_encode_port method

unittests.test_connection_unit.test_get_preview_ports()
    Test the get_preview_ports method

unittests.test_connection_unit.test_mark_face()
    Test the mark_face method

unittests.test_connection_unit.test_mark_tracking()
    Test the mark_tracking method

unittests.test_connection_unit.test_new_record()
    Test the new_record method

unittests.test_connection_unit.test_set_composite_mode()
    Test the set_composite_mode method

unittests.test_connection_unit.test_set_encode_mode()
    Test the set_encode_mode method

unittests.test_connection_unit.test_switch()
    Test the switch method
```

1.2.3 test_controller_unit Module

Unittests for Controller class in controller.py

```
class unittests.test_controller_unit.MockConnection(mode)
    Bases: object

    A class which mocks the Connection class

    adjust_pip(xpos, ypos, width, height)
        mock of adjust_pip

    click_video(xpos, ypos, width, height)
        mock of click_video

    get_audio_port()
        mock of get_audio_port
```



```

get_compose_port ()
    mock of get_compose_port

get_encode_port ()
    mock of get_encode_port

get_preview_ports ()
    mock of get_preview_ports

mark_face (face)
    mock of mark_face

mark_tracking (face)
    mock of mark_tracking

new_record ()
    mock of new_record

set_composite_mode (mode)
    mock of set_composite_mode

set_encode_mode (mode)
    mock of get_set_encode_mode

switch (channel, port)
    mock of switch

class unittests.test_controller_unit.TestAddress
    Bases: object

    Test the address parameter

    test_address_colon ()
        Test if address has no colon

    test_address_normal ()
        Test if address is valid

    test_address_null ()
        Test if address is null

class unittests.test_controller_unit.TestAdjustPIP
    Bases: object

    Test the adjust_pip method

    test_normal_unpack ()
        Test if valid

    test_unpack ()
        Test if unpack fails

class unittests.test_controller_unit.TestBusName
    Bases: object

    Test bus_name parameter

    test_normal ()
        Test when bus_name is not null

    test_normal_none ()
        Test when bus_name is null

```

```
class unittests.test_controller_unit.TestClickVideo
    Bases: object
    Test the click_video method
    test_normal_unpack ()
        Test if valid
    test_unpack ()
        Test if unpack fails

class unittests.test_controller_unit.TestEstablishConnection
    Bases: object
    Test the establish_connection method
    test_normal (monkeypatch)
        Test if the parameters are valid

class unittests.test_controller_unit.TestGetAudioPort
    Bases: object
    Test the get_audio_port method
    test_normal_unpack ()
        Test if valid
    test_unpack ()
        Test if unpack fails

class unittests.test_controller_unit.TestGetComposePort
    Bases: object
    Test the get_compose_port method
    test_normal_unpack ()
        Test when valid
    test_unpack ()
        Test when values cant unpack

class unittests.test_controller_unit.TestGetEncodePort
    Bases: object
    Test the get_encode_port method
    test_normal_unpack ()
        Test if valid
    test_unpack ()
        Test if unpack fails

class unittests.test_controller_unit.TestGetPreviewPorts
    Bases: object
    Test the get_preview_ports method
    test_normal_unpack ()
        Test if valid
    test_unpack ()
        Test if unpack fails

class unittests.test_controller_unit.TestInterface
    Bases: object
```

```

    Test the default_interface parameter

    test_interface_dot ()
        Test when the default_interface has <2 dots

    test_interface_none ()
        Test when the default_interface is null

    test_interface_normal ()
        Test when the interface is valid

class unittests.test_controller_unit.TestMarkFaces
    Bases: object

    Test the mark_face method

    test_normal ()
        Test if valid

class unittests.test_controller_unit.TestMarkTracking
    Bases: object

    Test the mark_tracking method

    test_normal ()
        Test if valid

class unittests.test_controller_unit.TestNewRecord
    Bases: object

    Test the new_record method

    test_normal_unpack ()
        Test if valid

    test_unpack ()
        Test if unpack fails

class unittests.test_controller_unit.TestObjectPath
    Bases: object

    Test object_path parameter

    test_object_path_blank ()
        Test when the object_path is null

    test_object_path_normal ()
        Test when object_path is valid

    test_object_path_slash ()
        Test when object_path doesn't have slash in start

class unittests.test_controller_unit.TestParsePreviewPorts
    Bases: object

    Test the parse_preview_ports class method

    test_normal ()
        Test if valid

    test_syntax_error ()
        Test if syntax error detected

    test_value_error ()
        Test if invalid

```

```
class unittests.test_controller_unit.TestSetCompositeMode
    Bases: object
    Test the set_composite_mode method

    test_normal_unpack()
        Test if valid

    test_unpack()
        Test if unpack fails

class unittests.test_controller_unit.TestSetEncodeMode
    Bases: object
    Test the set_encode_mode method

    test_normal_unpack()
        Test if valid

    test_unpack()
        Test if unpack fails

class unittests.test_controller_unit.TestSwitch
    Bases: object
    Test the switch method

    test_normal_unpack()
        Test if valid

    test_unpack()
        Test if unpack fails
```

1.2.4 test_helpers_unit Module

Unittests for helper.py

```
class unittests.test_helpers_unit.TestPreviewSinks
    Bases: object
    Test Preview Sinks

    class MockPreview (preview_port=3001)
        Bases: object
        A mock preview class

        end()
            End the preview

        run()
            Run the preview

    TestPreviewSinks.test_run (monkeypatch)
        Test running preview

    TestPreviewSinks.test_terminate_fail()
        Test terminating a preview when none exists

    TestPreviewSinks.test_terminate_normal (monkeypatch)
        Test terminating a preview when valid exists
```

```

class unittests.test_helpers_unit.TestPreviewSinksPreviewPort
    Bases: object
    Test preview_port parameter
    test_blank()
        Test when preview_port is blank
    test_invalid()
        Test when preview_port is not a integral value
    test_normal()
        Test when preview_port is valid
    test_range()
        Test when preview_port is out of range

class unittests.test_helpers_unit.TestTestSources
    Bases: object
    Test for testsources

    class MockAudioSrc (port, freq=110, wave=None)
        Bases: object
        A Mock audio source
        run()
            Run the source

    class TestTestSources.MockTest (pattern)
        Bases: object
        A mock test
        end()
            End the test

    class TestTestSources.MockTest2 (wave)
        Bases: object
        A mock audio source
        end()
            End the source

    class TestTestSources.MockVideoSrc (port, width=300, height=200, pattern=None, timeover-
                                         lay=False, clockoverlay=False)
        Bases: object
        A mock video source
        run()
            Run the mock video source

    TestTestSources.test_get_test_audio()
        Test get_test_audio

    TestTestSources.test_get_test_video()
        Test for get_test_video

    TestTestSources.test_new_test_audio (monkeypatch)
        Test new_test_audio

    TestTestSources.test_new_test_video (monkeypatch)
        Test for new_test_video

```

```
TestTestSources.test_terminate1_audio()
    Test terminate_audio multiple

TestTestSources.test_terminate1_video()
    Test terminate_video multiple

TestTestSources.test_terminate2_audio()
    Test terminate_audio none present

TestTestSources.test_terminate2_video()
    Test terminate_video none present

TestTestSources.test_terminate_index_error_audio()
    Test terminate_index_audio - invalid index

TestTestSources.test_terminate_index_error_video()
    Test for terminate_video

TestTestSources.test_terminate_index_normal_audio()
    Test terminate_index_audio valid index

TestTestSources.test_terminate_index_normal_video()
    Test terminate_index_video
```

```
class unittests.test_helpers_unit.TestTestSourcesAudioPort
    Bases: object
```

```
    Test for audio_port parameter
```

```
    test_invalid()
        Test when not integral value
```

```
    test_normal()
        Test when valid
```

```
    test_range()
        Test when out of range
```

```
class unittests.test_helpers_unit.TestTestSourcesVideoPort
    Bases: object
```

```
    Test for video_port parameter
```

```
    test_invalid()
        Test when not integral value
```

```
    test_normal()
        Test when valid
```

```
    test_range()
        Test when out of range
```

1.2.5 test_server_unit Module

Unittests for Server class in server.py

```
class unittests.test_server_unit.MockPopen(cmd, bufsize, shell)
    Bases: object
```

```
    Mock Popen method
```

```
    communicate()
        Mock communicate method of Popen
```

```

class unittests.test_server_unit.MockProcess(mode=True)
    Bases: object

    A mock process

    kill()
        Kill the mock process

    make_coverage()
        Dump coverage

    terminate()
        Terminate the mock process

class unittests.test_server_unit.TestAudioPort
    Bases: object

    Test for audio_port parameter

    test_invalid_audio_port_null()
        Test when the audio_port is null

    test_invalid_audio_port_range()
        Test when the audio port is not in range

    test_invalid_audio_port_type()
        Test when the audio port given is not a valid integral value

class unittests.test_server_unit.TestControlPort
    Bases: object

    Test the control_port parameter

    test_invalid_control_port_null()
        Test when the control port is null

    test_invalid_control_port_range()
        Test when the control port is not in range

    test_invalid_control_port_type()
        Test when the control port is not a valid integral value

class unittests.test_server_unit.TestKillTerminate
    Bases: object

    Test kill, terminate and gcov_flush methods

    test_gcov_flush_fail()
        Test when gcov_flush fails

    test_kill_fail()
        Test when kill fails

    test_no_process_gov_flush()
        Test when no process exists and gcov_flush is called

    test_no_process_kill()
        Test when no process exists and kill is called

    test_no_process_terminate()
        Test when no process exists and terminate is called

    test_terminate_fail()
        Test when terminate fails

```

class unittests.test_server_unit.**TestNormal**

Bases: object

Test the functioning of tests under valid conditions

test_kill (*monkeypatch*)

Test kill ServerProcessError

test_kill_cov (*monkeypatch*)

Test kill and gcov_flush ServerProcessError

test_make_coverage (*monkeypatch*)

Test dumping coverage

test_normal_gcov_flush (*monkeypatch*)

Test gcov_flush

test_normal_kill (*monkeypatch*)

Test kill when normally called

test_normal_terminate ()

Test terminal when normally called

test_terminate ()

Test terminate ServerProcessError

test_terminate_cov ()

Test terminate and gcov_flush ServerProcessError

class unittests.test_server_unit.**TestPath**

Bases: object

Test the path parameter

test_path_empty (*monkeypatch*)

Test if null path is given

test_path_provided_no_slash ()

Test if a path is provided

test_path_provided_slash ()

Test if a path is provided

class unittests.test_server_unit.**TestRecordFile**

Bases: object

Test the record_file parameter

test_record_file_false ()

Test if record file is False

test_record_file_invalid ()

Test when the record_file is invalid

test_record_file_slashes ()

Test when the record_file has forward slashes

test_record_file_true ()

Test if record file is True

test_record_file_valid ()

Test if record file is valid

test_record_file_valid_date ()

Test if record file is valid


```

    test_record_file_valid_space()
        Test if record file is valid and has a space
class unittests.test_server_unit.TestRun
    Bases: object

    Test running the server

    test_run()
        Test the run method

    test_run_process()
        Test _run_process method

    test_start_process_error(monkeypatch)
        Test _start_process method

    test_start_process_normal(monkeypatch)
        Test _start_process normally
class unittests.test_server_unit.TestVideoPort
    Bases: object

    Test for video_port parameter

    test_invalid_video_port_null()
        Test when the video_port is null

    test_invalid_video_port_range()
        Test when the video port is not in range

    test_invalid_video_port_type()
        Test when the video port given is not a valid integral value

```

1.2.6 test_testsource_unit Module

Unitests for testsource.py

```

class unittests.test_testsource_unit.MockPipeline
    Bases: object

    Mock Pipeline

    disable()
        Disable the pipeline

    pause()
        Pause the pipeline

    play()
        Play the pipeline
class unittests.test_testsource_unit.TestAudioSrcFreq
    Bases: object

    Test frequency parameter

    test_blank()
        Test when frequency is null

    test_invalid()
        Test when frequency is not a valid integral value

```

test_normal()
Test when frequency is valid

test_range()
Test when frequency is not in range (negative)

class unittests.test_testsource_unit.**TestAudioSrcPlay**
Bases: object

Test Audio Source options - play, pause, end

test_end()
End the audio source

test_pause()
Pause the audio source

test_run()
Run the audio source

class unittests.test_testsource_unit.**TestAudioSrcPort**
Bases: object

Test port parameter

test_blank()
Test when port is null

test_invalid()
Test when port is not a valid integral value

test_normal()
Test when port is valid

test_range()
Test when port is not in range

class unittests.test_testsource_unit.**TestAudioSrcWave**
Bases: object

Test wave parameter

test_invalid()
Test when wave is not a valid integral value

test_normal()
Test when wave is valid

test_range()
Test when wave is not in range

class unittests.test_testsource_unit.**TestBasePipeline**
Bases: object

Test Base Pipeline

test_disable (*monkeypatch*)
Test disable method

test_pause (*monkeypatch*)
Test pause method

test_play (*monkeypatch*)
Test play method

```

class unittests.test_testsource_unit.TestPreviewPlay
    Bases: object
    Test preview options - play, pause, end

    test_end()
        Test end method

    test_pause()
        Test pause method

    test_run()
        Test play method

class unittests.test_testsource_unit.TestPreviewPort
    Bases: object
    Test port parameter

    test_blank()
        Test when port is null

    test_invalid()
        Test when port is not a valid integral value

    test_normal()
        Test when port is valid

    test_range()
        Test when port is not in range

class unittests.test_testsource_unit.TestVideoPipeline
    Bases: object
    Test VideoPipeline

    test_permuate_time_clock_1()
        Test when timeoverlay False and clockoverlay False

    test_permuate_time_clock_2()
        test when timeoverlay False and clockoverlay True

    test_permuate_time_clock_3()
        Test when timeoverlay True and clockoverlay False

    test_permuate_time_clock_4()
        Test when timeoverlay True and clockoverlay True

class unittests.test_testsource_unit.TestVideoSrcClockOverlay
    Bases: object
    Test clockoverlay pattern

    test_fail()
        Test when clockoverlay is not boolean/valid

    test_normal()
        Test when clockoverlay is valid

class unittests.test_testsource_unit.TestVideoSrcHeight
    Bases: object
    Test for height parameter

```

```
test_blank()
    Test when the height is a null

test_invalid()
    Test when height is not a valid float value

test_non_positive()
    Test when height is non-positive

test_normal()
    Test when height is valid
```

```
class unittests.test_testsource_unit.TestVideoSrcPattern
    Bases: object
```

```
    Test the pattern parameter

test_invalid()
    Test when pattern is not a valid integer

test_normal()
    Test when pattern is valid

test_range()
    Test when pattern is not in range
```

```
class unittests.test_testsource_unit.TestVideoSrcPlay
    Bases: object
```

```
    Test Video Source options - play, pause, disable

test_end()
    Test end method

test_pause()
    Test pause method

test_run()
    Test run method
```

```
class unittests.test_testsource_unit.TestVideoSrcPort
    Bases: object
```

```
    Test the port parameter

test_blank()
    Test when the port is null

test_invalid()
    Test when the port is not a valid integral value

test_normal()
    Test when port is a valid value

test_range()
    Test when the port is not in range
```

```
class unittests.test_testsource_unit.TestVideoSrcTimeOverlay
    Bases: object
```

```
    Test timeoverlay parameter

test_fail()
    Test when timeoverlay is not boolean/valid
```

```

    test_normal ()
        Test when timeoverlay is valid
class unittests.test_testsource_unit.TestVideoSrcWidth
    Bases: object

    Test the width parameter

    test_blank ()
        Test when the width is null

    test_invalid ()
        Test when the width is not a valid float value

    test_non_positive ()
        Test when the width is non-positive

    test_normal ()
        Test when the width is valid

```

1.3 Integration Tests for gstswitch and Python-API

1.3.1 integrationtests Package

Integration tests for gst-switch and Python API

1.3.2 compare Module

1.3.3 generate_reference_frames Module

1.3.4 test_controller Module

1.3.5 test_helpers Module

Integration tests for TestSources, PreviewSinks in helpers.py

```

class integrationtests.test_helpers.TestTestSourcesPreviews
    Bases: object

    Test for TestSources and PreviewSinks

    NUM = 1

    add_audio_sources (num, audio_port)
        Add audio sources

    add_video_sources (num, video_port)
        Add a video source

    test_audio_sources ()
        Test audio sources

    test_video_sources ()
        Test video sources

```

1.3.6 test_server Module

Integration tests for Server in server.py

```
class integrationtests.test_server.TestServerStartStop
    Bases: object

    Test Starting and Stopping the Server Run Server and Stop multiple times

    NUM = 5

    startstop()
        Start and Stop the Server

    test_start_stop()
        Test Start and Stop the Server
```

Indices and tables

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

g

gstswitch.__init__, 1
gstswitch.connection, 1
gstswitch.controller, 3
gstswitch.exception, 6
gstswitch.helpers, 6
gstswitch.server, 7
gstswitch.testsource, 9

i

integrationtests.__init__, 25
integrationtests.test_helpers, 25
integrationtests.test_server, 26

u

unittests.__init__, 10
unittests.test_connection_unit, 10
unittests.test_controller_unit, 12
unittests.test_helpers_unit, 16
unittests.test_server_unit, 18
unittests.test_testsource_unit, 21

A

add_audio_sources() (integrationtests.test_helpers.TestTestSourcesPreviews method), 25

add_video_sources() (integrationtests.test_helpers.TestTestSourcesPreviews method), 25

address (gstswitch.connection.Connection attribute), 1

address (gstswitch.controller.Controller attribute), 4

adjust_pip() (gstswitch.connection.Connection method), 1

adjust_pip() (gstswitch.controller.Controller method), 4

adjust_pip() (unittests.test_controller_unit.MockConnection method), 12

audio_port (gstswitch.helpers.TestSources attribute), 6

audio_port (gstswitch.server.Server attribute), 8

AudioSrc (class in gstswitch.testsource), 10

B

BaseError, 6

bus_name (gstswitch.connection.Connection attribute), 2

bus_name (gstswitch.controller.Controller attribute), 4

C

call_sync() (unittests.test_connection_unit.MockConnection method), 10

click_video() (gstswitch.connection.Connection method), 2

click_video() (gstswitch.controller.Controller method), 4

click_video() (unittests.test_controller_unit.MockConnection method), 12

clockoverlay (gstswitch.testsource.VideoSrc attribute), 9

communicate() (unittests.test_server_unit.MockPopen method), 18

connect_dbus() (gstswitch.connection.Connection method), 2

Connection (class in gstswitch.connection), 1

CONNECTION_FLAGS (gstswitch.connection.Connection attribute), 1

ConnectionError, 6

ConnectionReturnError, 6

control_port (gstswitch.server.Server attribute), 8

Controller (class in gstswitch.controller), 3

D

default_interface (gstswitch.connection.Connection attribute), 2

default_interface (gstswitch.controller.Controller attribute), 4

disable() (unittests.test_testsource_unit.MockPipeline method), 21

E

end() (gstswitch.testsource.AudioSrc method), 10

end() (gstswitch.testsource.Preview method), 9

end() (gstswitch.testsource.VideoSrc method), 9

end() (unittests.test_helpers_unit.TestPreviewSinks.MockPreview method), 16

end() (unittests.test_helpers_unit.TestTestSources.MockTest method), 17

end() (unittests.test_helpers_unit.TestTestSources.MockTest2 method), 17

establish_connection() (gstswitch.controller.Controller method), 4

F

freq (gstswitch.testsource.AudioSrc attribute), 10

funcs (unittests.test_connection_unit.MockConnection attribute), 10

G

gcov_flush() (gstswitch.server.Server method), 8

generate_pattern() (gstswitch.testsource.VideoSrc class method), 9

generate_wave() (gstswitch.testsource.AudioSrc class method), 10

get_audio_port() (gstswitch.connection.Connection method), 2

get_audio_port() (gstswitch.controller.Controller method), 4

[get_audio_port\(\)](#) (unittests.test_controller_unit.MockConnection method), 12
[get_compose_port\(\)](#) (gstswitch.connection.Connection method), 2
[get_compose_port\(\)](#) (gstswitch.controller.Controller method), 4
[get_compose_port\(\)](#) (unittests.test_controller_unit.MockConnection method), 12
[get_encode_port\(\)](#) (gstswitch.connection.Connection method), 2
[get_encode_port\(\)](#) (gstswitch.controller.Controller method), 5
[get_encode_port\(\)](#) (unittests.test_controller_unit.MockConnection method), 13
[get_preview_ports\(\)](#) (gstswitch.connection.Connection method), 2
[get_preview_ports\(\)](#) (gstswitch.controller.Controller method), 5
[get_preview_ports\(\)](#) (unittests.test_controller_unit.MockConnection method), 13
[get_test_audio\(\)](#) (gstswitch.helpers.TestSources method), 6
[get_test_video\(\)](#) (gstswitch.helpers.TestSources method), 6
[gstswitch.__init__](#) (module), 1
[gstswitch.connection](#) (module), 1
[gstswitch.controller](#) (module), 3
[gstswitch.exception](#) (module), 6
[gstswitch.helpers](#) (module), 6
[gstswitch.server](#) (module), 7
[gstswitch.testsource](#) (module), 9

H

[height](#) (gstswitch.testsource.VideoSrc attribute), 9
[HOST](#) (gstswitch.testsource.AudioSrc attribute), 10
[HOST](#) (gstswitch.testsource.VideoSrc attribute), 9

I

[integrationtests.__init__](#) (module), 25
[integrationtests.test_helpers](#) (module), 25
[integrationtests.test_server](#) (module), 26
[InvalidIndexError](#), 6

K

[kill\(\)](#) (gstswitch.server.Server method), 8
[kill\(\)](#) (unittests.test_server_unit.MockProcess method), 19

M

[make_coverage\(\)](#) (gstswitch.server.Server class method), 8
[make_coverage\(\)](#) (unittests.test_server_unit.MockProcess method), 19

[mark_face\(\)](#) (gstswitch.connection.Connection method), 2
[mark_face\(\)](#) (gstswitch.controller.Controller method), 5
[mark_face\(\)](#) (unittests.test_controller_unit.MockConnection method), 13
[mark_tracking\(\)](#) (gstswitch.connection.Connection method), 3
[mark_tracking\(\)](#) (gstswitch.controller.Controller method), 5
[mark_tracking\(\)](#) (unittests.test_controller_unit.MockConnection method), 13
[MockConnection](#) (class in unittests.test_connection_unit), 10
[MockConnection](#) (class in unittests.test_controller_unit), 12
[MockPipeline](#) (class in unittests.test_testsource_unit), 21
[MockPopen](#) (class in unittests.test_server_unit), 18
[MockProcess](#) (class in unittests.test_server_unit), 18

N

[new_record\(\)](#) (gstswitch.connection.Connection method), 3
[new_record\(\)](#) (gstswitch.controller.Controller method), 5
[new_record\(\)](#) (unittests.test_controller_unit.MockConnection method), 13
[new_test_audio\(\)](#) (gstswitch.helpers.TestSources method), 6
[new_test_video\(\)](#) (gstswitch.helpers.TestSources method), 7
[NUM](#) (integrationtests.test_helpers.TestTestSourcesPreviews attribute), 25
[NUM](#) (integrationtests.test_server.TestServerStartStop attribute), 26

O

[object_path](#) (gstswitch.connection.Connection attribute), 3
[object_path](#) (gstswitch.controller.Controller attribute), 5

P

[parse_preview_ports\(\)](#) (gstswitch.controller.Controller class method), 5
[path](#) (gstswitch.server.Server attribute), 8
[PathError](#), 6
[pattern](#) (gstswitch.testsource.VideoSrc attribute), 9
[pause\(\)](#) (gstswitch.testsource.AudioSrc method), 10
[pause\(\)](#) (gstswitch.testsource.Preview method), 9
[pause\(\)](#) (gstswitch.testsource.VideoSrc method), 9
[pause\(\)](#) (unittests.test_testsource_unit.MockPipeline method), 21
[play\(\)](#) (unittests.test_testsource_unit.MockPipeline method), 21
[port](#) (gstswitch.testsource.AudioSrc attribute), 10
[port](#) (gstswitch.testsource.VideoSrc attribute), 9

Preview (class in `gstswitch.testsource`), 9
 preview_port (`gstswitch.helpers.PreviewSinks` attribute), 7
 preview_port (`gstswitch.testsource.Preview` attribute), 9
 PreviewSinks (class in `gstswitch.helpers`), 7

R

RangeError, 6
 record_file (`gstswitch.server.Server` attribute), 8
 run() (`gstswitch.helpers.PreviewSinks` method), 7
 run() (`gstswitch.server.Server` method), 8
 run() (`gstswitch.testsource.AudioSrc` method), 10
 run() (`gstswitch.testsource.Preview` method), 9
 run() (`gstswitch.testsource.VideoSrc` method), 9
 run() (`unittests.test_helpers_unit.TestPreviewSinks.MockPreviewSinks` method), 16
 run() (`unittests.test_helpers_unit.TestTestSources.MockAudioSrc` method), 17
 run() (`unittests.test_helpers_unit.TestTestSources.MockVideoSrc` method), 17
 running_tests_audio (`gstswitch.helpers.TestSources` attribute), 7
 running_tests_video (`gstswitch.helpers.TestSources` attribute), 7

S

Server (class in `gstswitch.server`), 7
 ServerProcessError, 6
 set_composite_mode() (`gstswitch.connection.Connection` method), 3
 set_composite_mode() (`gstswitch.controller.Controller` method), 5
 set_composite_mode() (`unittests.test_controller_unit.MockConnection` method), 13
 set_encode_mode() (`gstswitch.connection.Connection` method), 3
 set_encode_mode() (`gstswitch.controller.Controller` method), 5
 set_encode_mode() (`unittests.test_controller_unit.MockConnection` method), 13
 SLEEP_TIME (`gstswitch.server.Server` attribute), 8
 startstop() (`integrationtests.test_server.TestServerStartStop` method), 26
 switch() (`gstswitch.connection.Connection` method), 3
 switch() (`gstswitch.controller.Controller` method), 5
 switch() (`unittests.test_controller_unit.MockConnection` method), 13

T

terminate() (`gstswitch.helpers.PreviewSinks` method), 7
 terminate() (`gstswitch.server.Server` method), 8
 terminate() (`unittests.test_server_unit.MockProcess` method), 19

terminate_audio() (`gstswitch.helpers.TestSources` method), 7
 terminate_index_audio() (`gstswitch.helpers.TestSources` method), 7
 terminate_index_video() (`gstswitch.helpers.TestSources` method), 7
 terminate_video() (`gstswitch.helpers.TestSources` method), 7
 test_address_colon() (`unittests.test_connection_unit.TestAddress` method), 11
 test_address_colon() (`unittests.test_controller_unit.TestAddress` method), 13
 test_address_normal() (`unittests.test_connection_unit.TestAddress` method), 11
 test_address_normal() (`unittests.test_controller_unit.TestAddress` method), 13
 test_address_null() (`unittests.test_connection_unit.TestAddress` method), 11
 test_address_null() (`unittests.test_controller_unit.TestAddress` method), 13
 test_adjust_pip() (in module `unittests.test_connection_unit`), 12
 test_audio_sources() (in module `integrationtests.test_helpers.TestTestSourcesPreviews` method), 25
 test_bad_address() (`unittests.test_connection_unit.TestConnectDBus` method), 11
 test_bad_address2() (`unittests.test_connection_unit.TestConnectDBus` method), 11
 test_bad_address3() (`unittests.test_connection_unit.TestConnectDBus` method), 11
 test_blank() (`unittests.test_helpers_unit.TestPreviewSinksPreviewPort` method), 17
 test_blank() (`unittests.test_testsource_unit.TestAudioSrcFreq` method), 21
 test_blank() (`unittests.test_testsource_unit.TestAudioSrcPort` method), 22
 test_blank() (`unittests.test_testsource_unit.TestPreviewPort` method), 23
 test_blank() (`unittests.test_testsource_unit.TestVideoSrcHeight` method), 23
 test_blank() (`unittests.test_testsource_unit.TestVideoSrcPort` method), 24
 test_blank() (`unittests.test_testsource_unit.TestVideoSrcWidth` method), 25
 test_click_video() (in module `unittests.test_connection_unit`), 12
 test_disable() (`unittests.test_testsource_unit.TestBasePipeline` method), 22
 test_end() (`unittests.test_testsource_unit.TestAudioSrcPlay` method), 22
 test_end() (`unittests.test_testsource_unit.TestPreviewPlay` method), 23

[test_end\(\)](#) (unittests.test_testsource_unit.TestVideoSrcPlay method), [24](#)
[test_fail\(\)](#) (unittests.test_testsource_unit.TestVideoSrcClockOverlay method), [19](#)
[test_fail\(\)](#) (unittests.test_testsource_unit.TestVideoSrcTimeOverlay method), [24](#)
[test_gcov_flush_fail\(\)](#) (unittests.test_server_unit.TestKillTerminate method), [19](#)
[test_get_audio_port\(\)](#) (in module unittests.test_connection_unit), [12](#)
[test_get_compose_port\(\)](#) (in module unittests.test_connection_unit), [12](#)
[test_get_encode_port\(\)](#) (in module unittests.test_connection_unit), [12](#)
[test_get_preview_ports\(\)](#) (in module unittests.test_connection_unit), [12](#)
[test_get_test_audio\(\)](#) (unittests.test_helpers_unit.TestTestSources method), [17](#)
[test_get_test_video\(\)](#) (unittests.test_helpers_unit.TestTestSources method), [17](#)
[test_interface_dot\(\)](#) (unittests.test_connection_unit.TestInterface method), [11](#)
[test_interface_dot\(\)](#) (unittests.test_controller_unit.TestInterface method), [15](#)
[test_interface_none\(\)](#) (unittests.test_connection_unit.TestInterface method), [11](#)
[test_interface_none\(\)](#) (unittests.test_controller_unit.TestInterface method), [15](#)
[test_interface_normal\(\)](#) (unittests.test_connection_unit.TestInterface method), [11](#)
[test_interface_normal\(\)](#) (unittests.test_controller_unit.TestInterface method), [15](#)
[test_invalid\(\)](#) (unittests.test_helpers_unit.TestPreviewSinksPreviewPort method), [17](#)
[test_invalid\(\)](#) (unittests.test_helpers_unit.TestTestSourcesAudioPort method), [18](#)
[test_invalid\(\)](#) (unittests.test_helpers_unit.TestTestSourcesVideoPort method), [18](#)
[test_invalid\(\)](#) (unittests.test_testsource_unit.TestAudioSrcFreq method), [21](#)
[test_invalid\(\)](#) (unittests.test_testsource_unit.TestAudioSrcPort method), [22](#)
[test_invalid\(\)](#) (unittests.test_testsource_unit.TestAudioSrcWave method), [22](#)
[test_invalid\(\)](#) (unittests.test_testsource_unit.TestPreviewPort method), [23](#)
[test_invalid\(\)](#) (unittests.test_testsource_unit.TestVideoSrcHeight method), [24](#)
[test_invalid\(\)](#) (unittests.test_testsource_unit.TestVideoSrcPattern method), [24](#)
[test_invalid\(\)](#) (unittests.test_testsource_unit.TestVideoSrcPort method), [24](#)
[test_invalid\(\)](#) (unittests.test_testsource_unit.TestVideoSrcWidth method), [25](#)
[test_invalid_audio_port_null\(\)](#) (unittests.test_server_unit.TestAudioPort method), [19](#)
[test_invalid_audio_port_range\(\)](#) (unittests.test_server_unit.TestAudioPort method), [19](#)
[test_invalid_audio_port_type\(\)](#) (unittests.test_server_unit.TestAudioPort method), [19](#)
[test_invalid_control_port_null\(\)](#) (unittests.test_server_unit.TestControlPort method), [19](#)
[test_invalid_control_port_range\(\)](#) (unittests.test_server_unit.TestControlPort method), [19](#)
[test_invalid_control_port_type\(\)](#) (unittests.test_server_unit.TestControlPort method), [19](#)
[test_invalid_video_port_null\(\)](#) (unittests.test_server_unit.TestVideoPort method), [21](#)
[test_invalid_video_port_range\(\)](#) (unittests.test_server_unit.TestVideoPort method), [21](#)
[test_invalid_video_port_type\(\)](#) (unittests.test_server_unit.TestVideoPort method), [21](#)
[test_kill\(\)](#) (unittests.test_server_unit.TestNormal method), [20](#)
[test_kill_cov\(\)](#) (unittests.test_server_unit.TestNormal method), [20](#)
[test_kill_fail\(\)](#) (unittests.test_server_unit.TestKillTerminate method), [19](#)
[test_make_coverage\(\)](#) (unittests.test_server_unit.TestNormal method), [20](#)
[test_mark_face\(\)](#) (in module unittests.test_connection_unit), [12](#)
[test_mark_tracking\(\)](#) (in module unittests.test_connection_unit), [12](#)
[test_mock1\(\)](#) (unittests.test_connection_unit.TestConnectDBus method), [11](#)
[test_mock2\(\)](#) (unittests.test_connection_unit.TestConnectDBus method), [11](#)
[test_new_record\(\)](#) (in module unittests.test_connection_unit), [12](#)
[test_new_test_audio\(\)](#) (unittests.test_helpers_unit.TestTestSources method), [17](#)
[test_new_test_video\(\)](#) (unittests.test_helpers_unit.TestTestSources method), [17](#)
[test_no_process_gov_flush\(\)](#) (unittests.test_server_unit.TestKillTerminate method), [19](#)
[test_no_process_kill\(\)](#) (unittests.test_server_unit.TestKillTerminate method), [19](#)

test_no_process_terminate() (unittests.test_server_unit.TestKillTerminate method), 19	test_normal_terminate() (unittests.test_server_unit.TestNormal method), 20
test_non_positive() (unittests.test_testsource_unit.TestVideoSrcHeight method), 24	test_normal_unpack() (unittests.test_controller_unit.TestAdjustPIP method), 13
test_non_positive() (unittests.test_testsource_unit.TestVideoSrcWidth method), 25	test_normal_unpack() (unittests.test_controller_unit.TestClickVideo method), 14
test_normal() (unittests.test_connection_unit.TestBusName method), 11	test_normal_unpack() (unittests.test_controller_unit.TestGetAudioPort method), 14
test_normal() (unittests.test_controller_unit.TestBusName method), 13	test_normal_unpack() (unittests.test_controller_unit.TestGetComposePort method), 14
test_normal() (unittests.test_controller_unit.TestEstablishConnection method), 14	test_normal_unpack() (unittests.test_controller_unit.TestGetEncodePort method), 14
test_normal() (unittests.test_controller_unit.TestMarkFaces method), 15	test_normal_unpack() (unittests.test_controller_unit.TestGetPreviewPorts method), 14
test_normal() (unittests.test_controller_unit.TestMarkTracking method), 15	test_normal_unpack() (unittests.test_controller_unit.TestNewRecord method), 15
test_normal() (unittests.test_controller_unit.TestParsePreviewPorts method), 15	test_normal_unpack() (unittests.test_controller_unit.TestSetCompositeMod method), 16
test_normal() (unittests.test_helpers_unit.TestPreviewSinksPreviewPort method), 17	test_normal_unpack() (unittests.test_controller_unit.TestSetEncodeMode method), 16
test_normal() (unittests.test_helpers_unit.TestTestSourcesAudioPort method), 18	test_normal_unpack() (unittests.test_controller_unit.TestSwitch method), 16
test_normal() (unittests.test_helpers_unit.TestTestSourcesVideoPort method), 18	test_object_path_blank() (unittests.test_connection_unit.TestObjectPath method), 11
test_normal() (unittests.test_testsource_unit.TestAudioSrcFreq method), 21	test_object_path_blank() (unittests.test_controller_unit.TestObjectPath method), 15
test_normal() (unittests.test_testsource_unit.TestAudioSrcPort method), 22	test_object_path_normal() (unittests.test_connection_unit.TestObjectPath method), 12
test_normal() (unittests.test_testsource_unit.TestAudioSrcWidth method), 22	test_object_path_normal() (unittests.test_controller_unit.TestObjectPath method), 15
test_normal() (unittests.test_testsource_unit.TestPreviewPort method), 23	test_object_path_slash() (unittests.test_connection_unit.TestObjectPath method), 12
test_normal() (unittests.test_testsource_unit.TestVideoSrcClockOverlap method), 23	test_object_path_slash() (unittests.test_controller_unit.TestObjectPath method), 15
test_normal() (unittests.test_testsource_unit.TestVideoSrcHeight method), 24	test_path_empty() (unittests.test_server_unit.TestPath method), 20
test_normal() (unittests.test_testsource_unit.TestVideoSrcPattern method), 24	test_path_provided_no_slash() (unittests.test_server_unit.TestPath method), 20
test_normal() (unittests.test_testsource_unit.TestVideoSrcPort method), 24	test_path_provided_slash() (unittests.test_server_unit.TestPath method), 20
test_normal() (unittests.test_testsource_unit.TestVideoSrcTimeOffset method), 24	test_pause() (unittests.test_testsource_unit.TestAudioSrcPlay method), 22
test_normal() (unittests.test_testsource_unit.TestVideoSrcWidth method), 25	test_pause() (unittests.test_testsource_unit.TestBasePipeline method), 22
test_normal_gcov_flush() (unittests.test_server_unit.TestNormal method), 20	test_pause() (unittests.test_testsource_unit.TestPreviewPlay method), 23
test_normal_kill() (unittests.test_server_unit.TestNormal method), 20	test_pause() (unittests.test_testsource_unit.TestVideoSrcPlay method), 24
test_normal_none() (unittests.test_connection_unit.TestBusName method), 11	test_permute_time_clock_1() (unittests.test_testsource_unit.TestVideoPipeline method), 13

method), 23

test_permuate_time_clock_2()
(unittests.test_testsource_unit.TestVideoPipeline method), 23

test_permuate_time_clock_3()
(unittests.test_testsource_unit.TestVideoPipeline method), 23

test_permuate_time_clock_4()
(unittests.test_testsource_unit.TestVideoPipeline method), 23

test_play() (unittests.test_testsource_unit.TestBasePipeline method), 22

test_range() (unittests.test_helpers_unit.TestPreviewSinksPreviewPort method), 17

test_range() (unittests.test_helpers_unit.TestTestSourcesAudioPort method), 18

test_range() (unittests.test_helpers_unit.TestTestSourcesVideoPort method), 18

test_range() (unittests.test_testsource_unit.TestAudioSrcFreq method), 22

test_range() (unittests.test_testsource_unit.TestAudioSrcPort method), 22

test_range() (unittests.test_testsource_unit.TestAudioSrcWave method), 22

test_range() (unittests.test_testsource_unit.TestPreviewPort method), 23

test_range() (unittests.test_testsource_unit.TestVideoSrcPattern method), 24

test_range() (unittests.test_testsource_unit.TestVideoSrcPort method), 24

test_record_file_false() (unittests.test_server_unit.TestRecordFile method), 20

test_record_file_invalid()
(unittests.test_server_unit.TestRecordFile method), 20

test_record_file_slashes()
(unittests.test_server_unit.TestRecordFile method), 20

test_record_file_true() (unittests.test_server_unit.TestRecordFile method), 20

test_record_file_valid() (unittests.test_server_unit.TestRecordFile method), 20

test_record_file_valid_date()
(unittests.test_server_unit.TestRecordFile method), 20

test_record_file_valid_space()
(unittests.test_server_unit.TestRecordFile method), 20

test_run() (unittests.test_helpers_unit.TestPreviewSinks method), 16

test_run() (unittests.test_server_unit.TestRun method), 21

test_run() (unittests.test_testsource_unit.TestAudioSrcPlay method), 22

test_run() (unittests.test_testsource_unit.TestPreviewPlay method), 23

test_run() (unittests.test_testsource_unit.TestVideoSrcPlay method), 24

test_run_process() (unittests.test_server_unit.TestRun method), 21

test_set_composite_mode() (in module unittests.test_connection_unit), 12

test_set_encode_mode() (in module unittests.test_connection_unit), 12

test_start_process_error()
(unittests.test_server_unit.TestRun method), 21

test_start_process_normal()
(unittests.test_server_unit.TestRun method), 21

test_start_stop() (integrationtests.test_server.TestServerStartStop method), 26

test_switch() (in module unittests.test_connection_unit), 12

test_syntax_error() (unittests.test_controller_unit.TestParsePreviewPorts method), 15

test_terminate() (unittests.test_server_unit.TestNormal method), 20

test_terminate1_audio() (unittests.test_helpers_unit.TestTestSources method), 18

test_terminate1_video() (unittests.test_helpers_unit.TestTestSources method), 18

test_terminate2_audio() (unittests.test_helpers_unit.TestTestSources method), 18

test_terminate2_video() (unittests.test_helpers_unit.TestTestSources method), 18

test_terminate_cov() (unittests.test_server_unit.TestNormal method), 20

test_terminate_fail() (unittests.test_helpers_unit.TestPreviewSinks method), 16

test_terminate_fail() (unittests.test_server_unit.TestKillTerminate method), 19

test_terminate_index_error_audio()
(unittests.test_helpers_unit.TestTestSources method), 18

test_terminate_index_error_video()
(unittests.test_helpers_unit.TestTestSources method), 18

test_terminate_index_normal_audio()
(unittests.test_helpers_unit.TestTestSources method), 18

test_terminate_index_normal_video()
(unittests.test_helpers_unit.TestTestSources method), 18

test_terminate_normal() (unittests.test_helpers_unit.TestPreviewSinks method), 16

test_unpack() (unittests.test_controller_unit.TestAdjustPIP method), 13

test_unpack() (unittests.test_controller_unit.TestClickVideoTestInterface (class in unittests.test_connection_unit), 11
method), 14 TestInterface (class in unittests.test_controller_unit), 14

test_unpack() (unittests.test_controller_unit.TestGetAudioPortTestKillTerminate (class in unittests.test_server_unit), 19
method), 14 TestMarkFaces (class in unittests.test_controller_unit), 15

test_unpack() (unittests.test_controller_unit.TestGetComposePortTestMarkTracking (class in
method), 14 unittests.test_controller_unit), 15

test_unpack() (unittests.test_controller_unit.TestGetEncodePortTestNewRecord (class in unittests.test_controller_unit),
method), 14 15

test_unpack() (unittests.test_controller_unit.TestGetPreviewPortTestNormal (class in unittests.test_server_unit), 19
method), 14 TestObjectPath (class in unittests.test_connection_unit),

test_unpack() (unittests.test_controller_unit.TestNewRecord 11
method), 15 TestObjectPath (class in unittests.test_controller_unit), 15

test_unpack() (unittests.test_controller_unit.TestSetComposePortTestParsePreviewPorts (class in
method), 16 unittests.test_controller_unit), 15

test_unpack() (unittests.test_controller_unit.TestSetEncodeModeTestPath (class in unittests.test_server_unit), 20
method), 16 TestPreviewPlay (class in unittests.test_testsource_unit),

test_unpack() (unittests.test_controller_unit.TestSwitch 22
method), 16 TestPreviewPort (class in unittests.test_testsource_unit),

test_value_error() (unittests.test_controller_unit.TestParsePreviewPort33
method), 15 TestPreviewSinks (class in unittests.test_helpers_unit), 16

test_video_sources() (integrationtests.test_helpers.TestTestSourcesPreviews (class in
method), 25 unittests.test_helpers_unit), 16

TestAddress (class in unittests.test_connection_unit), 10 TestPreviewSinksPreviewPort (class in
TestAddress (class in unittests.test_controller_unit), 13 unittests.test_helpers_unit), 16

TestAdjustPIP (class in unittests.test_controller_unit), 13 TestRecordFile (class in unittests.test_server_unit), 20

TestAudioPort (class in unittests.test_server_unit), 19 TestRun (class in unittests.test_server_unit), 21

TestAudioSrcFreq (class in 26
unittests.test_testsource_unit), 21 TestServerStartStop (class in integrationtests.test_server),

TestAudioSrcPlay (class in
unittests.test_testsource_unit), 22 TestSetCompositeMode (class in
TestAudioSrcPort (class in unittests.test_testsource_unit),
22 unittests.test_controller_unit), 15

TestAudioSrcWave (class in
unittests.test_testsource_unit), 22 TestSetEncodeMode (class in
TestBasePipeline (class in unittests.test_testsource_unit),
22 unittests.test_controller_unit), 16

TestBusName (class in unittests.test_connection_unit), 11 TestSources (class in gstswitch.helpers), 6

TestBusName (class in unittests.test_controller_unit), 13 TestSwitch (class in unittests.test_controller_unit), 16

TestClickVideo (class in unittests.test_controller_unit),
13 TestTestSources (class in unittests.test_helpers_unit), 17

TestConnectDBus (class in
unittests.test_connection_unit), 11 TestTestSources.MockAudioSrc (class in
TestControlPort (class in unittests.test_server_unit), 19 unittests.test_helpers_unit), 17

TestEstablishConnection (class in
unittests.test_controller_unit), 14 TestTestSources.MockTest (class in
TestGetAudioPort (class in unittests.test_controller_unit),
14 unittests.test_helpers_unit), 17

TestGetComposePort (class in
unittests.test_controller_unit), 14 TestTestSources.MockTest2 (class in
TestGetEncodePort (class in
unittests.test_controller_unit), 14 unittests.test_helpers_unit), 17

TestGetPreviewPorts (class in
unittests.test_controller_unit), 14 TestTestSources.MockVideoSrc (class in
TestVideoPipeline (class in
unittests.test_testsource_unit), 23 unittests.test_helpers_unit), 17

TestVideoPort (class in unittests.test_server_unit), 21
TestVideoSrcClockOverlay (class in
unittests.test_testsource_unit), 23

TestVideoSrcHeight (class in unittests.test_testsource_unit), [23](#)
TestVideoSrcPattern (class in unittests.test_testsource_unit), [24](#)
TestVideoSrcPlay (class in unittests.test_testsource_unit), [24](#)
TestVideoSrcPort (class in unittests.test_testsource_unit), [24](#)
TestVideoSrcTimeOverlay (class in unittests.test_testsource_unit), [24](#)
TestVideoSrcWidth (class in unittests.test_testsource_unit), [25](#)
timeoverlay (gstswitch.testsource.VideoSrc attribute), [10](#)

U

unittests.__init__ (module), [10](#)
unittests.test_connection_unit (module), [10](#)
unittests.test_controller_unit (module), [12](#)
unittests.test_helpers_unit (module), [16](#)
unittests.test_server_unit (module), [18](#)
unittests.test_testsource_unit (module), [21](#)

V

video_port (gstswitch.helpers.TestSources attribute), [7](#)
video_port (gstswitch.server.Server attribute), [9](#)
VideoSrc (class in gstswitch.testsource), [9](#)

W

wave (gstswitch.testsource.AudioSrc attribute), [10](#)
width (gstswitch.testsource.VideoSrc attribute), [10](#)