SIGRun

Generated by Doxygen 1.8.15

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Data Structure Index

2.1 Data Structures

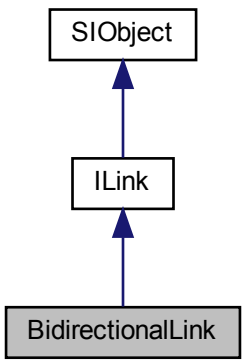
Here are the data structures with brief descriptions:

Chapter 3

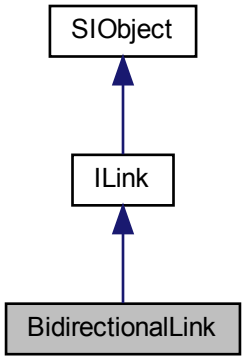
Data Structure Documentation

3.1 BidirectionalLink Class Reference

Inheritance diagram for BidirectionalLink:



Collaboration diagram for BidirectionalLink:



Public Member Functions

- **BidirectionalLink** (const std::shared_ptr< Region > &ra, const std::shared_ptr< Region > &rb, const std::string &aa, const std::string &ab)
- const LINK_TYPE & **type** () const override
- const std::shared_ptr< Region > & **sender_a** () const override
- const std::shared_ptr< Region > & **sender_b** () const override
- const std::shared_ptr< Region > & **receiver_a** () const override
- const std::shared_ptr< Region > & **receiver_b** () const override
- const std::shared_ptr< ExternalObject > & **external_sender_a** () const override
- const std::string & **attribute_a** () const override
- const std::string & **attribute_b** () const override
- virtual void **add_child** (std::shared_ptr< ILink > &link) override
- std::vector< std::shared_ptr< ILink > > & **children** () override
- const bool **is_external** () const override

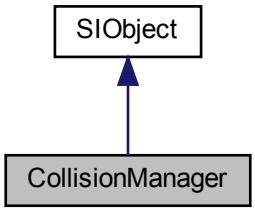
Additional Inherited Members

The documentation for this class was generated from the following files:

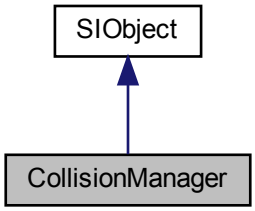
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/helpers/linking/Link.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/helpers/linking/Link.cpp

3.2 CollisionManager Class Reference

Inheritance diagram for CollisionManager:



Collaboration diagram for CollisionManager:



Public Member Functions

- void **collide** (std::vector< std::shared_ptr< Region >> &regions)
- void **handle_event_leave_on_deletion** (Region *deleted_region)

Friends

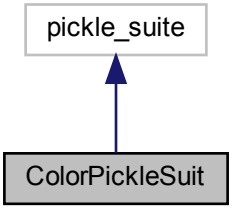
- class **Context**
- class **RegionManager**
- class **SIGRunCollisionManagerTest**

The documentation for this class was generated from the following files:

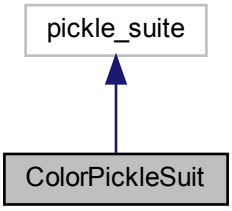
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/CollisionManager.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/CollisionManager.cpp

3.3 ColorPickleSuit Class Reference

Inheritance diagram for ColorPickleSuit:



Collaboration diagram for ColorPickleSuit:



Static Public Member Functions

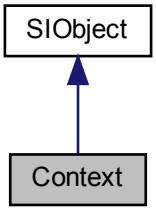
- static bp::tuple **getinitargs** (glm::vec4 &c)

The documentation for this class was generated from the following file:

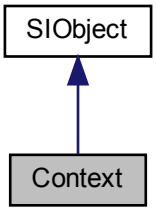
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.4 Context Class Reference

Inheritance diagram for Context:



Collaboration diagram for Context:



Public Member Functions

- void **begin** (const std::unordered_map< std::string, std::unique_ptr< bp::object >> &plugins, IRenderEngine *ire, IPhysicalEnvironment *ros, int argc, char **argv)
- void **end** ()
- RegionManager * **region_manager** ()
- InputManager * **input_manager** ()
- CollisionManager * **collision_manager** ()
- LinkingManager * **linking_manager** ()
- ExternalApplicationManager * **external_application_manager** ()
- JobSystem< void, 512 > * **job_system** ()
- TangibleManager * **tangible_manager** ()
- SpatialHashGrid * **spatial_hash_grid** ()
- QGraphicsView * **main_window** () const
- void **set_main_window** ()
- void **update** ()
- void **enable** (uint32_t what)
- void **disable** (uint32_t what)

- uint32_t **width** ()
- uint32_t **height** ()
- void **set_effect** (const std::string &target_uuid, const std::string &effect_name, bp::dict &kwargs)
- void **register_new_region** (const std::vector< glm::vec3 > &contour, const std::string &uuid, const bp::dict &kwargs)
- void **register_new_region_via_name** (const std::vector< glm::vec3 > &contour, const std::string &name, bool as_selector, bp::dict &kwargs)
- void **register_new_region_via_type** (const std::vector< glm::vec3 > &contour, int type, bp::dict &kwargs)
- void **register_region_via_class_object** (const std::vector< glm::vec3 > &contour, bp::object &clazz, bp::dict &kwargs)
- void **register_new_region_from_object** (const bp::object &object, const bp::dict &dict)
- void **register_link_event_emission** (const std::string &event_uuid, const std::string &sender_uuid, const std::string &sender_attribute, const bp::object &args)
- void **register_new_application_container** (uint64_t winid, uint64_t pid, const QString &window_name, const std::string &file_region_uuid)
- void **unregister_external_application** (const std::string &file_region_uuid)
- const std::unordered_map< std::string, bp::object > & **available_plugins** () const
- const bp::object & **plugin_by_name** (const std::string &name)
- const std::vector< std::string > & **available_plugins_names** ()
- const std::vector< std::string > & **excluded_plugins** ()
- std::unordered_map< std::string, std::shared_ptr< ExternalObject > > & **external_objects** ()
- const std::vector< std::string > & **conditional_variables** () const
- void **exclude_plugins** (const std::vector< std::string > &excluded_plugins)
- void **set_conditional_variables** (const std::vector< std::string > &conditionals)
- void **set_tangible_ip_address_and_port** (const std::string &ip, int port)
- void **set_pen_color** (int color)
- const int **pen_color** () const
- const std::unordered_map< std::string, bp::object > & **selected_effects_by_cursor_id** () const
- const IPhysicalEnvironment * **physical_environment** ()
- void **push_fps** (int actual, int target)
- void **click_mouse** (float x, float y)
- void **dbl_click_mouse** (float x, float y)
- void **set_file_system_root_folder** (const std::string &path)
- void **set_file_system_desktop_folder** (const std::string &path)
- std::string **file_system_root_folder** ()
- std::string **file_system_desktop_folder** ()

Static Public Member Functions

- static Context * **SIContext** ()

Data Fields

- std::ofstream **logfile**

Friends

- class **Core**
- class **SIGRunCollisionManagerTest**
- class **SIGRunLinkingManagerTest**
- class **SIGRunRegionTest**
- class **PySIPySIEffectTest**
- class **SIGRunLinkTest**

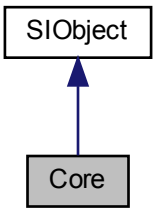
The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/Context.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/Context.cpp

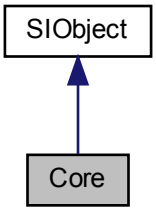
3.5 Core Class Reference

namespace shortening for python object integration

Inheritance diagram for Core:



Collaboration diagram for Core:



Public Member Functions

- ~Core ()
  destructor of Core class
- void start (char **argv, int argc, IRenderEngine *ire, IPhysicalEnvironment *ros)
  entry point of core SIGRun initialization
- void stop ()
  exit SIGRun core

Protected Member Functions

- Core ()
  constructor of Core class
- void retrieve_available_plugins (std::unordered_map< std::string, std::unique_ptr< bp::object >> &plugins, const std::string &plugin_path)

retrieve all available plugins before launching SIGRun environment

- void **prepare_plugin_loading** (std::vector< std::tuple< std::string, std::string >> &to_load, const std::vector< std::tuple< std::string, std::string >> &files, const std::string &plugin_path, const std::string &path_addition, Scripting &script)
- void **load_plugins** (std::unordered_map< std::string, std::unique_ptr< bp::object >> &plugins, const std::vector< std::tuple< std::string, std::string >> &to_load, Scripting &script)
- void **process_plugin_file** (std::vector< std::tuple< std::string, std::string >> &to_load, const std::string &path_addition, const std::tuple< std::string, std::string > &file, Scripting &script)
- void **filesystem_operations** (const std::string &loaded_path, const std::string &path, const std::string &name, const std::string &source)
- void **create_transpiled_plugin_files** (const std::string &source, const std::string &loaded_path, const std::string &name)
- void **copy_qml_and_res** (const std::string &path, const std::string &loaded_path)

Friends

- class **SIGRun**
- class **SIGRunTest**
- class **SIGRunRegionTest**
- class **SIGRunCoreTest**

3.5.1 Detailed Description

namespace shortening for python object integration

SIObject Central Core class registered as SIObject

This class initiates all subsystems required for the SIGRun environment. This class collects all available pulgins first. Second, it launches the SI context and other subsystems. This class is registered as SIObject meta type. This class conctructor is declared private to disable use by external application programmers. Therefore, the friend keyword is used to internally expose the class.

3.5.2 Constructor & Destructor Documentation

3.5.2.1 ~Core()

```
Core::~Core ( )
```

destructor of Core class

Shut down the SIGRun environment.

3.5.2.2 Core()

```
Core::Core ( )  [protected], [default]
```

constructor of Core class

Constructor which registers instance as an SIObject. Specify, which Logging capabilities are desired.

3.5.3 Member Function Documentation

3.5.3.1 retrieve_available_plugins()

```
void Core::retrieve_available_plugins (
            std::unordered_map< std::string, std::unique_ptr< bp::object >> & plugins,
            const std::string & plugin_path )  [protected]
```

retrieve all available plugins before launching SIGRun environment

Load all plugins in the plugin path of the SIGRun environment.

Parameters

plugins	a mutable reference to a std::unordered map with std::string as key and a std::shared_ptr of boost::python::objects as values which is the out parameter
plugin_path	a std::string which contains the path to the root folder of all plugin files

See also

Scripting::Scripting
PluginCollector::PluginCollector

3.5.3.2 start()

```
void Core::start (
            char ** argv,
            int argc,
            IRenderEngine * ire,
            IPhysicalEnvironment * ros )
```

entry point of core SIGRun initialization

Entry point of SIGRun's core which performs Plugin loading and initializes the SI Context.

Parameters

in	*argv*	
in	*argc*	
in	*ire*	
in	*ros*	

Returns

void

3.5.3.3 stop()

```
void Core::stop ( )
```

exit SIGRun core

Initiate the shutdown of the SIGRun core.

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/Core.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/Core.cpp

3.6 CrashDump Class Reference

Static Public Member Functions

- static void **dump_crash_information** (int32_t signal)

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/log/CrashDump.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/log/CrashDump.cpp

3.7 E Class Reference

Static Public Member Functions

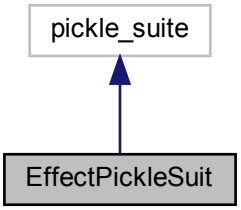
- static void **generate** ()

The documentation for this class was generated from the following files:

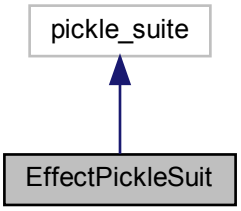
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/e/E.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/e/E.cpp

3.8 EffectPickleSuit Class Reference

Inheritance diagram for EffectPickleSuit:



Collaboration diagram for EffectPickleSuit:



Static Public Member Functions

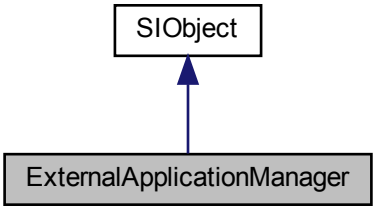
- static bp::tuple **getstate** (bp::object o)
- static void **setstate** (bp::object o, bp::tuple state)
- static bool **getstate_manages_dict** ()

The documentation for this class was generated from the following file:

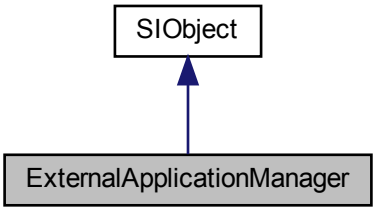
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.9 ExternalApplicationManager Class Reference

Inheritance diagram for ExternalApplicationManager:



Collaboration diagram for ExternalApplicationManager:



Public Member Functions

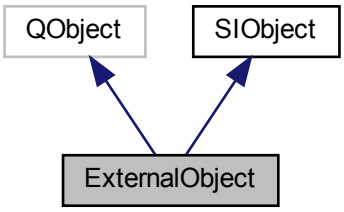
- **ExternalApplicationManager** (double process_winid_fetch_sleep_time_ms=DEFAULT_PROCESS_W↩ INID_FETCH_SLEEP_TIME_MS, double process_winid_fetch_timeout_ms=DEFAULT_PROCESS_WINI↩ D_FETCH_TIMEOUT_MS)
- void **launch_application** (const std::string &uuid, const std::string &file_path, std::shared_ptr< Region > &reg, const std::string &application_name="")
- void **launch_standard_application** (const std::string &uuid, const std::string &file_path)
- void **terminate_application** (const std::string &uuid)
- void **set_process_winid_fetch_sleep_time_ms** (double time)
- void **set_process_winid_fetch_timeout_ms** (double time)
- double **process_winid_fetch_sleep_time_ms** ()
- double **process_winid_fetch_timeout_ms** ()
- uint32_t **process_winid_fetch_iterations** ()

The documentation for this class was generated from the following files:

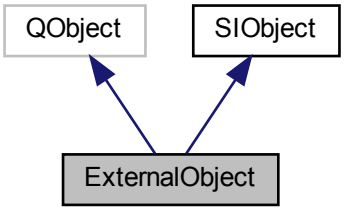
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/ExternalApplication↩ Manager.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/ExternalApplication↩ Manager.cpp

3.10 ExternalObject Class Reference

Inheritance diagram for ExternalObject:



Collaboration diagram for ExternalObject:



Public Types

- enum **ExternalObjectType** { **MOUSE** = 0, **FINGER** = 1, **TANGIBLE** = 2, **APPLICATION** = 3 }

Public Member Functions

- **ExternalObject** (const ExternalObjectType &type)
- const ExternalObjectType & **type** () const
- Q_SIGNAL void **LINK_SIGNAL** (const std::string &uuid_event, const std::string &uuid_sender, const std↩ ::string &source_cap, const bp::tuple &args)
- const std::string & **uuid** () const

Data Fields

- union {
  struct {
  } **mouse**
  struct {
  QWidget ∗ **window**
  uint64_t **pid**
  char ∗ **file_uuid**
  } **external_application**
  } **embedded_object**

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/helpers/input/External↩ Object.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/helpers/input/External↩ Object.cpp

3.11 Helper Class Reference

Static Public Member Functions

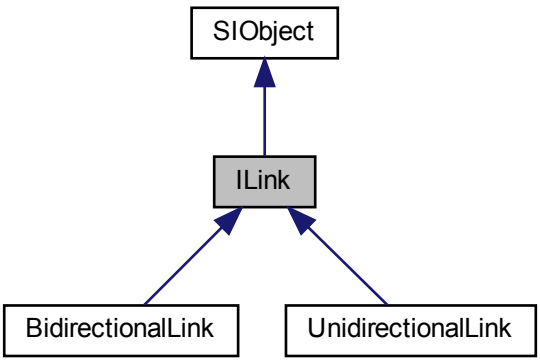
- static **std::vector** (const std::vector< glm::vec3 > &points)
- static **std::vector** (const std::vector< glm::vec3 > &points, float theta)
- static **std::vector** (const std::vector< glm::vec3 > &points, float size)
- static **std::vector** (const std::vector< glm::vec3 > &points)
- static **float** (const std::vector< glm::vec3 > &points, const Template &t, float a, float b, float threshold)
- static **float** (const std::vector< glm::vec3 > &points, const Template &t, float theta)
- static **glm::vec3** (const std::vector< glm::vec3 > &points)
- static std::vector< glm::vec3 > **bounding_box** (const std::vector< glm::vec3 > &points)
- static float **path_distance** (const std::vector< glm::vec3 > &pts1, const std::vector< glm::vec3 > &pts2)
- static float **distance** (const glm::vec3 &a, const glm::vec3 &b)

The documentation for this class was generated from the following file:

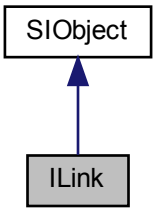
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/Dollar1GestureRecognizer.hpp

3.12 ILink Class Reference

Inheritance diagram for ILink:



Collaboration diagram for ILink:



Public Types

- enum **LINK_TYPE** { **UD**, **BD** }

Public Member Functions

- virtual const LINK_TYPE & **type** () const =0
- virtual const std::shared_ptr< Region > & **sender_a** () const =0
- virtual const std::shared_ptr< Region > & **sender_b** () const =0
- virtual const std::shared_ptr< Region > & **receiver_a** () const =0
- virtual const std::shared_ptr< Region > & **receiver_b** () const =0
- virtual const std::shared_ptr< ExternalObject > & **external_sender_a** () const =0
- virtual const std::string & **attribute_a** () const =0

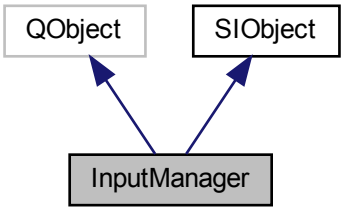
- virtual const std::string & **attribute_b** () const =0
- virtual const bool **is_external** () const =0
- virtual void **add_child** (std::shared_ptr< ILink > &link)=0
- virtual std::vector< std::shared_ptr< ILink > > & **children** ()=0

The documentation for this class was generated from the following file:

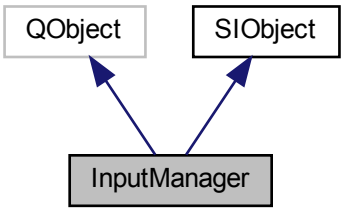
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/helpers/linking/Link.↩ hpp

3.13 InputManager Class Reference

Inheritance diagram for InputManager:



Collaboration diagram for InputManager:



Data Structures

- struct MouseWheelAngles

Public Member Functions

- bool **eventFilter** (QObject ∗watched, QEvent ∗event) override
- void **update** ()
- void **press_key** (uint32_t key_id)
- void **release_key** (uint32_t key_id)
- void **press_mouse_button** (uint32_t button_id)
- void **release_mouse_button** (uint32_t button_id)
- bool **is_key_down** (uint32_t key_id)
- bool **is_key_pressed** (uint32_t key_id)
- bool **is_mouse_down** (uint32_t button_id)
- bool **is_mouse_pressed** (uint32_t button_id)
- const glm::vec2 & **mouse_coords** () const
- const glm::vec2 & **previous_mouse_coords** () const
- const MouseWheelAngles **mouse_wheel_angles** ()
- const bool **is_double_click** ()

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/InputManager.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/InputManager.cpp

3.14 IPhysicalEnvironment Class Reference

Public Member Functions

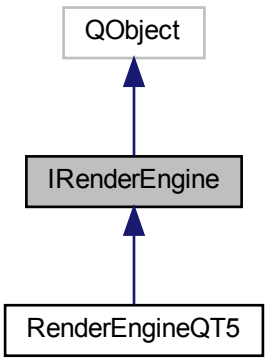
- virtual void **start** (int argc, char ∗∗argv)=0
- virtual void **stop** ()=0
- virtual void **send** (const std::string &msg) const =0
- virtual void **update** ()=0

The documentation for this class was generated from the following file:

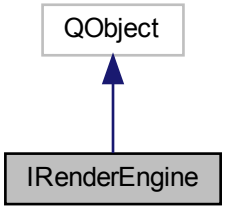
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/network/IPhysicalEnvironment.hpp

3.15 IRenderEngine Class Reference

Inheritance diagram for IRenderEngine:



Collaboration diagram for IRenderEngine:



Public Member Functions

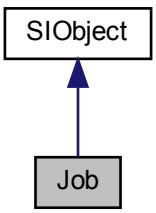
- virtual void **start** (uint32_t width, uint32_t height, uint32_t target_fps=60)=0
- virtual void **run** ()=0
- virtual void **pause** ()=0
- virtual void **stop** ()=0
- virtual void **set_cursor_stroke_width_by_cursor_id** (const std::string &cursor_id, int stroke_width)=0
- virtual void **set_cursor_stroke_color_by_cursor_id** (const std::string &cursor_id, const glm::vec4 &color)=0
- virtual void **disable_anti_aliasing** ()=0
- virtual void **enable_anti_aliasing** (uint32_t samplng_factor)=0

The documentation for this class was generated from the following file:

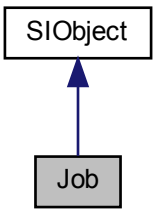
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/rendering/IRenderEngine.hpp

3.16 Job Class Reference

Inheritance diagram for Job:



Collaboration diagram for Job:



Public Types

- enum **PRIORITY** { **HIGH**, **NORMAL**, **LOW** }

Public Member Functions

- **Job** (const std::function< void()> &job_func, const PRIORITY &priority=PRIORITY::NORMAL)
- **Job** (const std::function< void(const JobDispatchArgs &args)> &job_func, const PRIORITY &priority=PRIORITY::NORMAL)
- void **execute** ()
- void **operator()** ()
- void **execute** (const JobDispatchArgs &args)
- void **operator()** (const JobDispatchArgs &args)

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/parallel/Job.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/parallel/Job.cpp

3.17 JobDispatchArgs Struct Reference

Public Member Functions

- **JobDispatchArgs** (uint32_t ji, uint32_t gi)

Data Fields

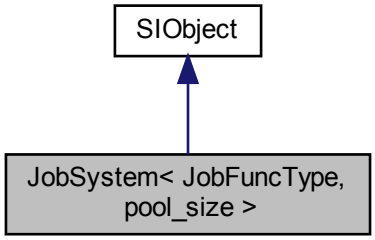
- uint32_t **jobIndex**
- uint32_t **groupIndex**

The documentation for this struct was generated from the following file:

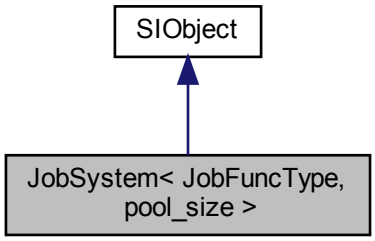
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/parallel/Job.hpp

3.18 JobSystem< JobFuncType, pool_size > Class Template Reference

Inheritance diagram for JobSystem< JobFuncType, pool_size >:



Collaboration diagram for JobSystem< JobFuncType, pool_size >:



Public Member Functions

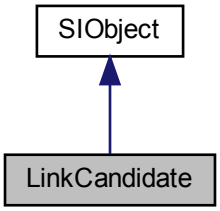
- void **poll** ()
- void **stop** ()
- void **execute** (const std::function< void()> &func)
- bool **is_busy** ()
- void **wait** ()
- void **dispatch** (uint32_t job_count, uint32_t group_size, const std::function< void(const JobDispatchArgs &args)> &func)

The documentation for this class was generated from the following file:

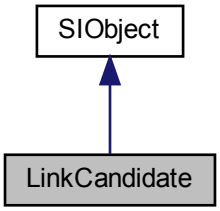
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/parallel/JobSystem.hpp

3.19 LinkCandidate Class Reference

Inheritance diagram for LinkCandidate:



Collaboration diagram for LinkCandidate:



Public Member Functions

- **LinkCandidate** (const std::string &_sender, const std::string &_sender_attrib, const std::string &_recv, const std::string &_recv_attrib)
- const bool **operator==** (const LinkCandidate &other) const
- const bool **operator !=** (const LinkCandidate &other) const

Data Fields

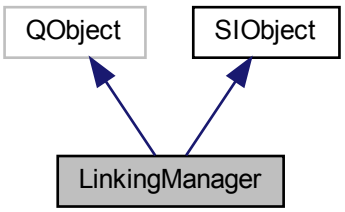
- std::string **sender**
- std::string **sender_attrib**
- std::string **recv**
- std::string **recv_attrib**

The documentation for this class was generated from the following file:

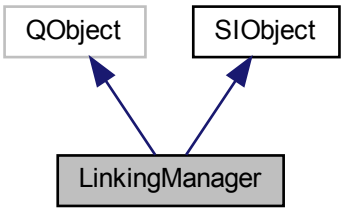
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/helpers/linking/Link↩ Candidate.hpp

3.20 LinkingManager Class Reference

Inheritance diagram for LinkingManager:



Collaboration diagram for LinkingManager:



Public Member Functions

- bool **add_link** (const std::shared_ptr< Region > &ra, const std::string &aa, const std::shared_ptr< Region > &rb, const std::string &ab, const ILink::LINK_TYPE &type)
- void **add_link** (std::shared_ptr< ExternalObject > &eo, std::shared_ptr< Region > &a, const std::string &ea, const std::string &aa)
- void **remove_link** (const std::shared_ptr< Region > &ra, const std::string &aa, const std::shared_ptr< Region > &rb, const std::string &ab, const ILink::LINK_TYPE &type)
- void **remove_link** (std::shared_ptr< ExternalObject > &eo, std::shared_ptr< Region > &a, const std::string &ea, const std::string &aa)
- bool **is_linked** (const std::shared_ptr< Region > &ra, const std::string &aa, const std::shared_ptr< Region > &rb, const std::string &ab, const ILink::LINK_TYPE &type)
- bool **is_linked** (const std::string &ra_uuid, const std::string &aa, const std::string &rb_uuid, const std::string &ab, const ILink::LINK_TYPE &type)
- bool **is_linked** (const std::shared_ptr< ExternalObject > &eo, const std::string &ea, const std::shared_ptr< Region > &ra, const std::string &aa)
- bool **is_linked** (const std::string &eo_uuid, const std::string &ea, const std::string &rb_uuid, const std::string &ab)

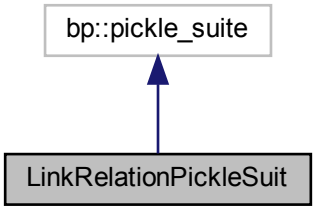
- void **emit_link_event** (std::shared_ptr< Region > &a, const std::string &attr_a)
- void **register_link_event_emission** (const std::string &event_uuid, const std::string &sender_uuid, const std::string &sender_attribute, const bp::object &args)
- void **perform_link_events** ()
- void **remove_links_by_indices** (std::vector< uint32_t > &indices)
- const std::vector< std::shared_ptr< ILink > > & **links** () const
- const uint64_t **num_links** () const
- void **update_linking_candidates** (std::vector< LinkCandidate > &relations, const std::string &source)
- void **remove_all_partaking_linking_relations** (const std::string &source)

The documentation for this class was generated from the following files:

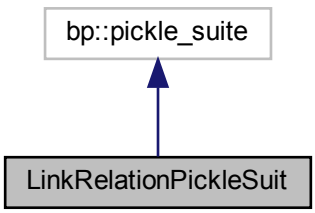
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/LinkingManager.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/LinkingManager.cpp

3.21 LinkRelationPickleSuit Class Reference

Inheritance diagram for LinkRelationPickleSuit:



Collaboration diagram for LinkRelationPickleSuit:



Static Public Member Functions

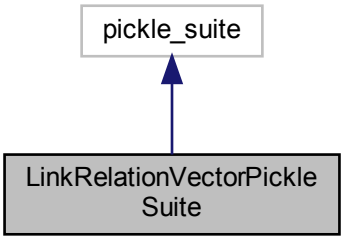
- static bp::tuple **getinitargs** (LinkCandidate &lc)

The documentation for this class was generated from the following file:

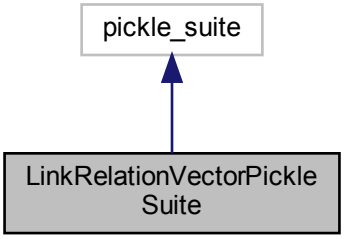
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.22 LinkRelationVectorPickleSuite Class Reference

Inheritance diagram for LinkRelationVectorPickleSuite:



Collaboration diagram for LinkRelationVectorPickleSuite:



Static Public Member Functions

- static bp::tuple **getinitargs** (std::vector< LinkCandidate > &v)

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.23 Log Class Reference

Log class serving as central logging functionality for easy logging data output.

`#include <Log.hpp>`

Public Types

- enum LOG_LEVEL {
  INFO_LEVEL = 0b00001, **WARN_LEVEL** = 0b00010, **DEBUG_LEVEL** = 0b00100, **ERROR_LEVEL** = 0b01000,
  UNDEFINED_LEVEL = 0b10000 }

  enum for log level selection modelled as a bitfield
- enum MODE { **NONE** = 0, **CONSOLE** = 1, **FILE** = 2 }

  enum for log mode selection modelled as a bitfield
- enum SHOW_TYPE {
  HIDDEN = 0, **INFO** = 1, **WARN** = 2, **DEBUG** = 4,
  ERROR = 8, **UNDEFINED** = 16 }

  enum for log show type selection modelled as a bitfield

Static Public Member Functions

- template<typename T >
  static void **log** (const std::string &origin, const T &what, uint16_t level, const std::string &type, const std::string &file="", const std::string &func="", const std::string &line="")
- static void **log** (const std::string &origin, const char *what, uint16_t level, const std::string &type, const std::string &file="", const std::string &func="", const std::string &line="")
- static void log (const std::string &origin, const std::string &what, uint16_t level, const std::string &type, const std::string &file="", const std::string &func="", const std::string &line="")

  central logging function outputting log messages according to its params
- static void set_log_file_path (const std::string &path)
- static std::string log_level (uint16_t log_level)

  return the level of a log message as tag according to its id
- static std::string time ()

  return current system time with milliseconds precision
- static void **quench** (const std::string &target)
- static void **unquench** (const std::string &target)
- static const std::vector< std::string > & **messages** ()

Static Public Attributes

- static std::string log_file_path = Log::PATH_DEFAULT
- static int16_t SHOW = -1

  the integer variable containing which log messages are outputted based on their tag
- static uint16_t WHERE = 0
- static bool __DEBUG__ = false

  the flag which is required to be set to true if the logging system is required to be used.
- static std::vector< std::string > **QUENCHED** = std::vector<std::string>()

3.23.1 Detailed Description

Log class serving as central logging functionality for easy logging data output.

This class serves as the central knot for all output operations in terms of log messages. This class is a static class featuring no ctor or dtor. Shortcut macros make the access to this class logging functionality more easier.

See also

DEBUG(what, log_mode)
WARN(what, log_mode)
ERROR(what, log_mode)
INFO(what, log_mode)
UNDEFINED(what, log_mode)

3.23.2 Member Enumeration Documentation

3.23.2.1 LOG_LEVEL

`enum Log::LOG_LEVEL`

enum for log level selection modelled as a bitfield

The log level describes which tag is assigned to a log message.

3.23.2.2 MODE

`enum Log::MODE`

enum for log mode selection modelled as a bitfield

The log mode describes where a log message is outputted. A mode is ignored if it is not specified. Due to the enum being modelled as a bitfield, users can use the | operator to selectively enable modes for logging output. Example for enabling printing to stdout as well as to a file: int mode = CONSOLE | FILE;

3.23.2.3 SHOW_TYPE

`enum Log::SHOW_TYPE`

enum for log show type selection modelled as a bitfield

The log show type describes which log messages are outputted based on their tags. Tags which are not specified are ignored. Due to the enum being modelled as a bitfield, users can use the | operator to selectively enable tags for logging output. Example for enabling DEBUG and WARN tags without the INFO tag: int loglevel = WARN | DEBUG

3.23.3 Member Function Documentation

3.23.3.1 log()

```
void Log::log (
            const std::string & origin,
            const std::string & what,
            uint16_t level,
            const std::string & type,
            const std::string & file = "",
            const std::string & func = "",
            const std::string & line = "" )  [static]
```

central logging function outputting log messages according to its params

This is the central logging function of SIGRun. It requires to be called from class which are registered as SIObject. The parameters of this function, besides what (log message), configure the way the message is outputted. This static method is easier accessible via the shortcut macros.

Parameters

what	a std::string containing the log message
level	an integer containing the id of the desired tag
logging_flags	an integer containing where the log message is to be outputted
type	a std::string containing the description of the functions caller via an SIObject
file	a std::string containing the name of the file in which the log call is implemented
func	a std::string containing the name of the function in which the log call was issued
line	a std::string containing the number of the line of the file in which the log call is implemented

See also

DEBUG(what, log_mode)
WARN(what, log_mode)
ERROR(what, log_mode)
INFO(what, log_mode)
UNDEFINED(what, log_mode)
SIObject

3.23.3.2 log_level()

```
std::string Log::log_level (
            uint16_t log_level )  [static]
```

return the level of a log message as tag according to its id

Retrieves the level of a log message according to the value of the parameter which is compared to the Log::LOG_LEVEL enum/bitfield.

Parameters

log_level	an integer containing the id of the desired tag

Returns

a std::string which contains a human readable version of the desired tag

3.23.3.3 set_log_file_path()

```
void Log::set_log_file_path (
            const std::string & path )  [static]
```

set the path of the file for logging output Set the value of the static variable log_file_path to the value of the given parameter to specifiy the file path of the log output.

Parameters

path	a std::string containing the desired file path for logging to files

3.23.3.4 time()

```
std::string Log::time ( )  [static]
```

return current system time with milliseconds precision

Compute current system time with milliseconds precision. Format the date data to yyyy-MM-dd hh:mm↩ :ss.<milliseconds>. Concatenate the date data to a std::string.

Returns

a std::string containing the formatted date data

3.23.4 Field Documentation

3.23.4.1 __DEBUG__

```
bool Log::__DEBUG__ = false  [static]
```

the flag which is required to be set to true if the logging system is required to be used.

This flag is the center of enabling (**DEBUG** is set to true) or disabling (**DEBUG** is set to false) the entire logging system.

3.23.4.2 log_file_path

```
std::string Log::log_file_path = Log::PATH_DEFAULT  [static]
```

actual path to logfile

3.23.4.3 SHOW

```
int16_t Log::SHOW = -1  [static]
```

the integer variable containing which log messages are outputted based on their tag

This integer variable regulates which log messages are outputted, according to their tags.

3.23.4.4 WHERE

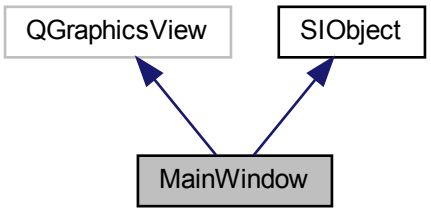
```
uint16_t Log::WHERE = 0  [static]
```

The documentation for this class was generated from the following files:

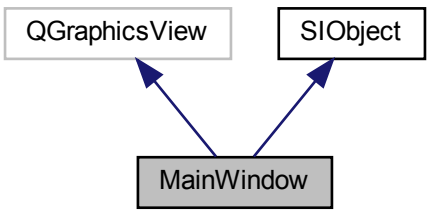
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/log/Log.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/log/Log.cpp

3.24 MainWindow Class Reference

Inheritance diagram for MainWindow:



Collaboration diagram for MainWindow:



Public Member Functions

- **MainWindow** (uint32_t width, uint32_t height, uint32_t target_fps)
- void **pause** ()
- void **loop** ()
- void **set_cursor_stroke_color_by_cursor_id** (const std::string &cursor_id, const glm::vec4 &color)
- void **set_cursor_stroke_width_by_cursor_id** (const std::string &cursor_id, int stroke_width)
- QQmlEngine * **engine** ()

Data Fields

- bool **d_is_running** = false

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/window/MainWindow.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/window/MainWindow.cpp

3.25 MapExposure< T > Class Template Reference

MapExposure class providing the interface for exposing STL maps to the python3 bindings (PySI) in a pythonic way.

`#include <MapExposure.hpp>`

Public Types

- typedef T::key_type K
- typedef T::mapped_type V

Static Public Member Functions

- static V & **get** (T &x, K const &i)
- static void **set** (T &x, K const &i, V const &v)
- static void **del** (T &x, K const &i)
- static constexpr bool **in** (T const &x, K const &i)
- static bp::list **keys** (T const &x)
- static bp::list **values** (T const &x)
- static bp::list **items** (T const &x)
- static constexpr int32_t **index** (T const &x, K const &k)

3.25.1 Detailed Description

template<typename T>
class MapExposure< T >

MapExposure class providing the interface for exposing STL maps to the python3 bindings (PySI) in a pythonic way.

Template Parameters

T	the STL map to be exposed

3.25.2 Member Typedef Documentation

3.25.2.1 K

```
template<typename T >
typedef T::key_type MapExposure< T >::K
```

Template Parameters

K	the type of key the STL map T contains

3.25.2.2 V

```
template<typename T >
typedef T::mapped_type MapExposure< T >::V
```

Template Parameters

V	the type of value the STL map T contains

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/MapExposure.hpp

3.26 MapExposurePartialContour Class Reference

MapExposurePartialContour class providing the interface for exposing a STL map to the python3 bindings (PySI) in a pythonic way which provides the functionality to add partial contours, i.e. regions which are in the process of being drawn, based on the uuid of the used cursor.

```
#include <MapExposure.hpp>
```

Static Public Member Functions

- static boost::shared_ptr< std::unordered_map< std::string, std::vector< glm::vec3 > > > init (const bp::dict &dict=bp::dict())

  the constructor of the MapExposurePartialContour
- static void set (std::unordered_map< std::string, std::vector< glm::vec3 >> &self, const std::string &key, const std::vector< glm::vec3 > &points)

  a member function which adds a new key value pair to the given MapExposurePartialContour self
- static std::string repr (std::unordered_map< std::string, std::vector< glm::vec3 >> &self)

  a member function which returns a string which contains the data of the map in readable way which can also be used from python

3.26.1 Detailed Description

MapExposurePartialContour class providing the interface for exposing a STL map to the python3 bindings (PySI) in a pythonic way which provides the functionality to add partial contours, i.e. regions which are in the process of being drawn, based on the uuid of the used cursor.

3.26.2 Member Function Documentation

3.26.2.1 init()

```
static boost::shared_ptr<std::unordered_map<std::string, std::vector<glm::vec3> > > Map↩
ExposurePartialContour::init (
            const bp::dict & dict = bp::dict() )  [inline], [static]
```

the constructor of the MapExposurePartialContour

Parameters

in	*dict*	the python dictionary which contains cursor uuids as keys and partial contours as values.

Returns

a boost::shared_ptr<std::unordered_map<std::string, std::vector<glm::vec3>>> which stores the data of the given dictionary dict

3.26.2.2 repr()

```
static std::string MapExposurePartialContour::repr (
            std::unordered_map< std::string, std::vector< glm::vec3 >> & self )  [inline],
[static]
```

a member function which returns a string which contains the data of the map in readable way which can also be used from python

Parameters

in	*self*	the map to be represented by a string

Returns

the string containing the representation of the map

3.26.2.3 set()

```
static void MapExposurePartialContour::set (
            std::unordered_map< std::string, std::vector< glm::vec3 >> & self,
            const std::string & key,
            const std::vector< glm::vec3 > & points )  [inline], [static]
```

a member function which adds a new key value pair to the given MapExposurePartialContour self

Parameters

in,out	*self*	a map to which a new key/value pair is to be added
in	*key*	a std::string which contains the uuid of a cursor used for drawing a region
in	*points*	a std::vector<glm::vec3> which contains the points of the partial contour which is drawn

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/MapExposure.hpp

3.27 MapExposureString2_String2FunctionMap_Map Class Reference

MapExposureString2_String2FunctionMap_Map class providing the interface for exposing a STL map to the python3 bindings (PySI) in a pythonic way which provides the functionality of a map which uses capabilities as keys to other maps which use event keywords, such as on_enter, on_continous, and on_leave for collision events, or another capbility for linking events as keys to the event function.

`#include <MapExposure.hpp>`

Static Public Member Functions

- static boost::shared_ptr< std::unordered_map< std::string, std::unordered_map< std::string, bp::object > > > init (const bp::dict &dict=bp::dict())

  the constructor of the MapExposureString2_String2FunctionMap_Map
- static void set (std::unordered_map< std::string, std::unordered_map< std::string, bp::object >> &self, const std::string &key, const bp::dict &dict)

  a member function which adds a new key value pair to the given MapExposureString2Function self
- static std::string repr (std::unordered_map< std::string, std::unordered_map< std::string, bp::object >> &self)

  a member function which returns a string which contains the data of the map self in readable way which can also be used from python

3.27.1 Detailed Description

MapExposureString2_String2FunctionMap_Map class providing the interface for exposing a STL map to the python3 bindings (PySI) in a pythonic way which provides the functionality of a map which uses capabilities as keys to other maps which use event keywords, such as on_enter, on_continous, and on_leave for collision events, or another capbility for linking events as keys to the event function.

3.27.2 Member Function Documentation

3.27.2.1 init()

```
static boost::shared_ptr<std::unordered_map<std::string, std::unordered_map<std::string, bp↩
::object> > > MapExposureString2_String2FunctionMap_Map::init (
            const bp::dict & dict = bp::dict() )  [inline], [static]
```

the constructor of the MapExposureString2_String2FunctionMap_Map

Parameters

in	*dict*	the python dictionary which contains capabilities as keys and and an inner map as value which contains the event keywords or capabilities as keys and the event function as values.

Returns

a boost::shared_ptr<std::unordered_map<std::string, std::unordered_map<std::string, bp::object>>> which stores the data of the given dictionary dict

3.27.2.2 repr()

```
static std::string MapExposureString2_String2FunctionMap_Map::repr (
            std::unordered_map< std::string, std::unordered_map< std::string, bp::object >>
& self )  [inline], [static]
```

a member function which returns a string which contains the data of the map self in readable way which can also be used from python

Parameters

in	*self*	the map to be represented by a string

Returns

the string containing the representation of the map

3.27.2.3 set()

```
static void MapExposureString2_String2FunctionMap_Map::set (
            std::unordered_map< std::string, std::unordered_map< std::string, bp::object >>
& self,
            const std::string & key,
            const bp::dict & dict )  [inline], [static]
```

a member function which adds a new key value pair to the given MapExposureString2Function self

Parameters

in,out	*self*	a map to which a new key/value pair is to be added
in	*key*	a std::string which contains the capability of an event
in	*dict*	a boost::python::dict which contains the key/value pairs of event keywords or capabilities and event functions.

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/MapExposure.hpp

3.28 MapExposureString2Function Class Reference

MapExposureString2Function class providing the interface for exposing a STL map to the python3 bindings (PySI) in a pythonic way which provides the functionality to add a region event capaility as key and the event function as value.

```
#include <MapExposure.hpp>
```

Static Public Member Functions

- static boost::shared_ptr< std::unordered_map< std::string, bp::object > > init (const bp::dict &dict=bp::dict())

  the constructor of the MapExposureString2Function
- static void set (std::unordered_map< std::string, bp::object > &self, const std::string &key, const bp::object &function)

  a member function which adds a new key value pair to the given MapExposureString2Function self
- static std::string repr (std::unordered_map< std::string, bp::object > &self)

  a member function which returns a string which contains the data of the map in readable way which can also be used from python

3.28.1 Detailed Description

MapExposureString2Function class providing the interface for exposing a STL map to the python3 bindings (PySI) in a pythonic way which provides the functionality to add a region event capaility as key and the event function as value.

3.28.2 Member Function Documentation

3.28.2.1 init()

```
static boost::shared_ptr<std::unordered_map<std::string, bp::object> > MapExposureString2↩
Function::init (
            const bp::dict & dict = bp::dict() )  [inline], [static]
```

the constructor of the MapExposureString2Function

Parameters

in	*dict*	the python dictionary which contains capabilities as keys and event functions as values.

Returns

a boost::shared_ptr<std::unordered_map<std::string, bp::object>> which stores the data of the given dictionary dict

3.28.2.2 repr()

```
static std::string MapExposureString2Function::repr (
            std::unordered_map< std::string, bp::object > & self )  [inline], [static]
```

a member function which returns a string which contains the data of the map in readable way which can also be used from python

Parameters

in	*self*	the map to be represented by a string

Returns

the string containing the representation of the map

3.28.2.3 set()

```
static void MapExposureString2Function::set (
            std::unordered_map< std::string, bp::object > & self,
            const std::string & key,
            const bp::object & function )  [inline], [static]
```

a member function which adds a new key value pair to the given MapExposureString2Function self

Parameters

in,out	*self*	a map to which a new key/value pair is to be added
in	*key*	a std::string which contains the capability of an event
in	*function*	a boost::python::object which contains the function of an event

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/MapExposure.hpp

3.29 InputManager::MouseWheelAngles Struct Reference

Data Fields

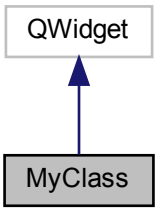
- float **px**
- float **degrees**

The documentation for this struct was generated from the following file:

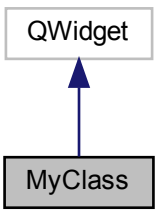
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/InputManager.hpp

3.30 MyClass Class Reference

Inheritance diagram for MyClass:



Collaboration diagram for MyClass:



Public Member Functions

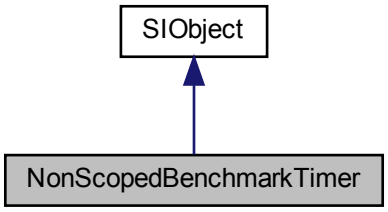
- **MyClass** (QString command, QWidget ∗parent=0)

The documentation for this class was generated from the following files:

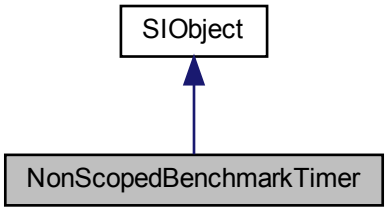
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sitools/sitools.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sitools/sitools.cpp

3.31 NonScopedBenchmarkTimer Class Reference

Inheritance diagram for NonScopedBenchmarkTimer:



Collaboration diagram for NonScopedBenchmarkTimer:



Public Member Functions

- **NonScopedBenchmarkTimer** (NonScopedBenchmarkTimer const &)=delete
- void **operator=** (NonScopedBenchmarkTimer const &)=delete
- void **mark_start** ()
- void **mark_stop** ()

Static Public Member Functions

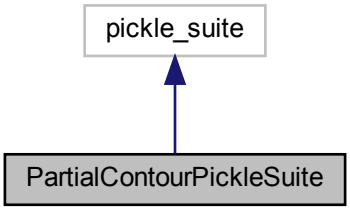
- static NonScopedBenchmarkTimer & **instance** ()

The documentation for this class was generated from the following file:

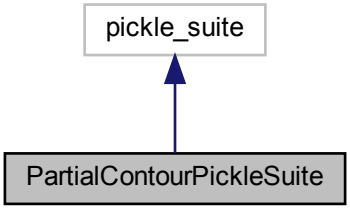
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/Benchmark.hpp

3.32 PartialContourPickleSuite Class Reference

Inheritance diagram for PartialContourPickleSuite:



Collaboration diagram for PartialContourPickleSuite:



Static Public Member Functions

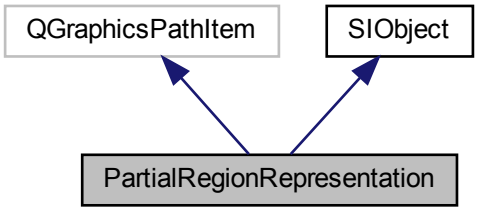
- static bp::tuple **getinitargs** (std::unordered_map< std::string, std::vector< glm::vec3 >> m)

The documentation for this class was generated from the following file:

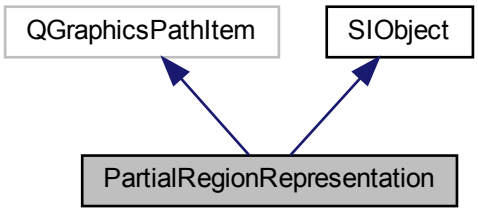
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.33 PartialRegionRepresentation Class Reference

Inheritance diagram for PartialRegionRepresentation:



Collaboration diagram for PartialRegionRepresentation:



Public Member Functions

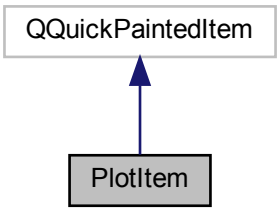
- **PartialRegionRepresentation** (const std::string &id, const std::vector< glm::vec3 > &source_contour, int stroke_width=4, const glm::vec4 &stroke_color=glm::vec4(72, 79, 81, 255))
- void **update** (const std::vector< glm::vec3 > &path)
- const std::string & **id** () const

The documentation for this class was generated from the following files:

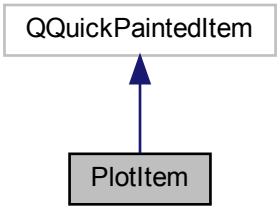
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/region/PartialRegionRepresentation.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/region/PartialRegionRepresentation.cpp

3.34 PlotItem Class Reference

Inheritance diagram for PlotItem:



Collaboration diagram for PlotItem:



Public Member Functions

- **PlotItem** (QQuickItem *parent=nullptr)
- QImage **image** () const
- void **setImage** (const QImage &image)
- void **paint** (QPainter *painter)

Properties

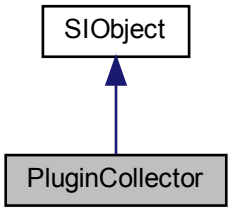
- QImage **image**

The documentation for this class was generated from the following files:

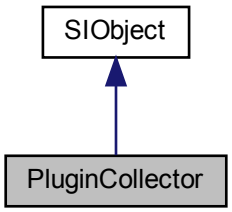
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/rendering/qml/items/PlotItem.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/rendering/qml/items/PlotItem.cpp

3.35 PluginCollector Class Reference

Inheritance diagram for PluginCollector:



Collaboration diagram for PluginCollector:



Public Member Functions

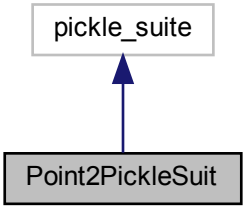
- void **collect** (const std::string &rel_path, std::vector< std::tuple< std::string, std::string >> &files)

The documentation for this class was generated from the following files:

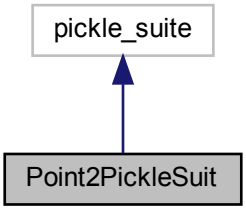
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/plugin/PluginCollector.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/plugin/PluginCollector.cpp

3.36 Point2PickleSuit Class Reference

Inheritance diagram for Point2PickleSuit:



Collaboration diagram for Point2PickleSuit:



Static Public Member Functions

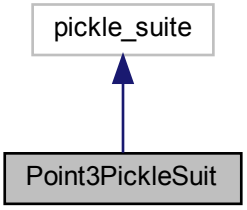
- static bp::tuple **getinitargs** (glm::vec2 &p)

The documentation for this class was generated from the following file:

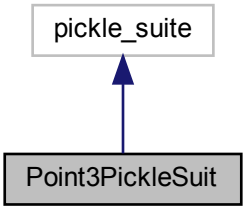
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.37 Point3PickleSuit Class Reference

Inheritance diagram for Point3PickleSuit:



Collaboration diagram for Point3PickleSuit:



Static Public Member Functions

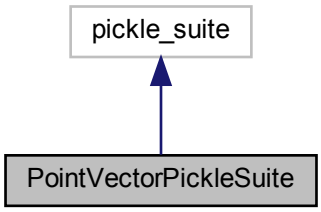
- static bp::tuple **getinitargs** (glm::vec3 &p)

The documentation for this class was generated from the following file:

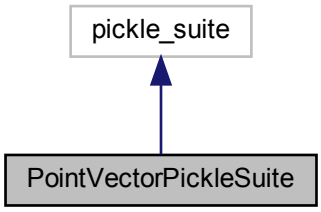
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.38 PointVectorPickleSuite Class Reference

Inheritance diagram for PointVectorPickleSuite:



Collaboration diagram for PointVectorPickleSuite:



Static Public Member Functions

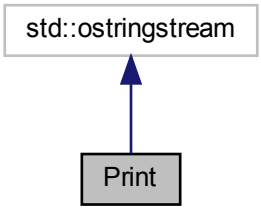
- static bp::tuple **getinitargs** (std::vector< glm::vec3 > &v)

The documentation for this class was generated from the following file:

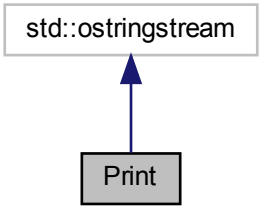
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.39 Print Class Reference

Inheritance diagram for Print:



Collaboration diagram for Print:



Static Public Member Functions

- template<typename TupleType , typename FunctionType >
  static void **for_each** (TupleType &&, FunctionType, std::integral_constant< size_t, std::tuple_size< typename std::remove_reference< TupleType >::type >::value >)
- template<std::size_t I, typename TupleType , typename FunctionType , typename = typename std::enable_if<I != std::tuple_size<typename std::remove_reference<TupleType>::type>::value>::type>
  static void **for_each** (TupleType &&t, FunctionType f, std::integral_constant< size_t, I >)
- template<typename TupleType , typename FunctionType >
  static void **for_each** (TupleType &&t, FunctionType f)
- template<typename T >
  static std::string **_print** (const std::vector< std::vector< T >> &v)
- template<typename T >
  static std::string **_print** (const std::vector< T > &v)
- template<typename T1 , typename T2 >
  static std::string **_print** (const std::map< T1, T2 > &map)
- static std::string **_print** (const QString &qs)
- static std::string **_print** (const QVariant &qv)

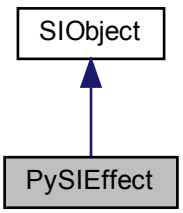
- template<typename T >
  static T **_print** (T &p)
- static std::string **_print** (int p)
- static std::string **_print** (int16_t p)
- static std::string **_print** (int64_t p)
- static std::string **_print** (uint32_t p)
- static std::string **_print** (uint16_t p)
- static std::string **_print** (uint64_t p)
- static std::string **_print** (float p)
- static std::string **_print** (double p)
- static std::string **_print** (char p)
- static std::string **_print** (int8_t p)
- static std::string **_print** (uint8_t p)
- static std::string **_print** (const glm::vec3 &p)
- static std::string **_print** (const glm::vec2 &p)
- static std::string **_print** (const glm::ivec4 &p)
- static std::string **_print** (const TangibleObjectMessage *msg)
- static std::string **_print** (RegionTransform *tform)
- static std::string **_print** (const bp::dict &d)
- static std::string **_print** (const bp::str str)
- static std::string **_print** (const bp::tuple t)
- template<class... Args>
  static void **print** (Args &&... args)

The documentation for this class was generated from the following files:

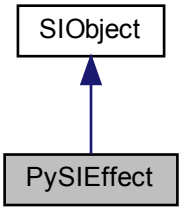
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/debug/Print.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/debug/Print.cpp

3.40 PySIEffect Class Reference

Inheritance diagram for PySIEffect:



Collaboration diagram for PySIEffect:



Public Member Functions

- **PySIEffect** (const std::vector< glm::vec3 > &contour, const std::string &uuid, const std::string &tex_path, const bp::dict &kwargs)
- void **__set_data__** (const std::string &key, const bp::object &value, const uint32_t type, const bp::dict &data_kwargs)
- void **__embed_file_standard_appliation_into_context__** (const std::string &uuid, const std::string &path)
- void **__destroy_embedded_file_standard_appliation_in_context__** (const std::string &uuid)
- void **__signal_deletion__** ()
- void **__signal_deletion_by_uuid__** (const std::string &uuid)
- void **__assign_effect__** (const std::string &sender, const std::string &effect_name, const std::string &effect↩ _display_name, bp::dict &kwargs)
- void **__emit_linking_action__** (const std::string &sender, const std::string &linking_action, const bp::object &args)
- void **__set_cursor_stroke_width_by_cursorid__** (const std::string &cursor_id, int width)
- void **__set_cursor_stroke_color_by_cursorid__** (const std::string &cursor_id, const glm::vec4 &color)
- void **__on_destroy__** ()
- void **__click_mouse__** (float x, float y)
- void **__dbl_click_mouse__** (float x, float y)
- void **__create_region__** (const std::vector< glm::vec3 > &contour, const std::string &name, bool as_↩ selector, bp::dict &kwargs)
- void **__create_region__** (const bp::list &contour, const std::string &name, bool as_selector, bp::dict &kwargs)
- void **__create_region__** (const bp::list &contour, int effect_type, bp::dict &kwargs)
- void **__create_region__** (const bp::object &contour, const bp::dict &qml)
- void **__create_region__** (const bp::list &contour, bp::object &clazz, bp::dict &kwargs)
- void **__current_tangible_selection__** (const std::string &effect_to_assign, const std::string &effect_↩ display_name, const std::string &effect_texture, bp::dict &kwargs)
- void **__add_multiple_regions__** (const bp::list &contours, const std::string &effect_name, bp::dict &kwargs)
- bp::list **__current_regions__** ()
- bp::list **__excluded_plugins__** ()
- bp::list **__conditional_variables__** ()
- void **__move_hard__** (float x, float y)
- void **__set_drawing_additions__** (const bp::list &drawing_additions)
- bp::list **__drawing_additions__** ()
- bp::dict **__qml_data_keys_and_types__** ()
- void **__update_transform__** (int32_t delta_x, int32_t delta_y)
- bp::list **__logger_messages__** ()
- bp::object **__data__** (const std::string &key, const uint32_t type)

- bp::tuple **__context_dimensions__** ()
- std::vector< std::string > **__available_plugins_by_name__** ()
- std::vector< glm::vec3 > **get_shape** ()
- const std::vector< glm::vec3 > & **original_shape** ()
- void **set_shape** (const std::vector< glm::vec3 > &shape)
- std::vector< std::vector< std::string > > **get_collisions** ()
- void **set_collisions** (const std::vector< std::vector< std::string >> &collisions)
- const int32_t **x** () const
- const int32_t **y** () const
- const int32_t **width** () const
- const int32_t **height** () const
- const int32_t **visualization_width** () const
- const int32_t **visualization_height** () const
- const uint32_t **effect_type** () const
- const float **scale** () const
- const float **angle_degrees** () const
- const std::string & **name** () const
- const std::string & **qml_path** () const
- const std::string & **source** () const
- const std::string & **uuid** () const
- void **set_mouse_pressed_capability** (uint32_t btn, bool active)
- bool **has_mouse_pressed_capability** (uint32_t btn)
- bool **is_flagged_for_deletion** ()
- bool **is_border_present** ()
- bool **visible** ()
- const bool **has_data_changed** () const
- bp::dict **__selected_effects_by_cursor_id__** ()
- std::vector< std::string > & **regions_for_registration** ()
- bp::list & **regions_for_registration_kwargs** ()
- std::vector< LinkCandidate > & **link_relations** ()
- std::vector< glm::vec3 > & **contour** ()
- void **set_aabb** (const std::vector< glm::vec3 > &aabb)
- std::vector< glm::vec3 > & **aabb** ()
- std::vector< std::vector< std::vector< glm::vec3 > > > & **drawing_additions** ()
- const glm::vec4 & **color** () const
- std::unordered_map< std::string, bp::object > & **attr_link_emit** ()
- std::unordered_map< std::string, std::unordered_map< std::string, bp::object > > & **attr_link_recv** ()
- std::unordered_map< std::string, std::unordered_map< std::string, bp::object > > & **cap_collision_emit** ()
- std::unordered_map< std::string, std::unordered_map< std::string, bp::object > > & **cap_collision_recv** ()
- std::unordered_map< std::string, std::vector< glm::vec3 > > & **partial_region_contours** ()
- void **set_data** (const QMap< QString, QVariant > &data)
- const QMap< QString, QVariant > & **data** ()
- bool **evaluate_enveloped** () const
- bool **is_enveloped** () const
- void **__notify__** (const bp::object &msg, const int type)

Data Fields

- float **d_x** = 0
- float **d_y** = 0
- int32_t **d_visualization_width** = 0
- int32_t **d_visualization_height** = 0
- int32_t **d_width** = 0
- int32_t **d_height** = 0

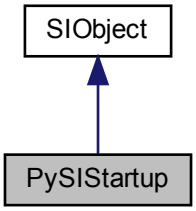
- uint32_t **d_effect_type** = SI_TYPE_CUSTOM
- int32_t **d_transform_x** = 0
- int32_t **d_transform_y** = 0
- float **d_scale** = 1.0f
- float **d_angle_deg** = 0.0
- std::string **d_name** = ""
- std::string **d_uuid** = ""
- std::string **d_qml_path** = ""
- std::string **d_source** = ""
- bool **d_is_left_mouse_clicked** = false
- bool **d_is_right_mouse_clicked** = false
- bool **d_is_middle_mouse_clicked** = false
- bool **d_is_double_clicked** = false
- bool **d_recompute_mask** = false
- bool **d_with_border** = false
- bool **d_visible** = true
- bool **d_evaluate_enveloped** = false
- bool **d_is_enveloped** = false
- bp::list **d_enveloped_by**
- float **mouse_wheel_angle_degrees** = 0.0
- float **mouse_wheel_angle_px** = 0.0
- bool **d_flagged_for_deletion** = false
- bool **d_is_resampling_enabled** = true
- std::vector< std::string > **d_regions_marked_for_registration**
- bp::list **d_regions_marked_for_registration_kwargs**
- std::vector< LinkCandidate > **d_link_relations**
- std::vector< glm::vec3 > **d_contour**
- std::vector< glm::vec3 > **d_original_contour**
- std::vector< glm::vec3 > **d_aabb**
- std::vector< std::vector< std::string > > **d_collisions**
- std::vector< std::vector< std::vector< glm::vec3 > > > **d_drawing_additions**
- glm::vec4 **d_color**
- glm::vec4 **d_border_color**
- int **d_border_width**
- std::unordered_map< std::string, bp::object > **d_cap_link_emit**
- std::unordered_map< std::string, std::unordered_map< std::string, bp::object > > **d_cap_link_recv**
- std::unordered_map< std::string, std::unordered_map< std::string, bp::object > > **d_cap_collision_emit**
- std::unordered_map< std::string, std::unordered_map< std::string, bp::object > > **d_cap_collision_recv**
- std::unordered_map< std::string, std::vector< glm::vec3 > > **d_partial_regions**
- bool **d_data_changed**

The documentation for this class was generated from the following files:

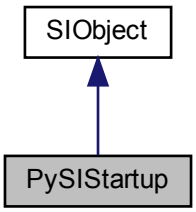
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/PySIEffect.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/PySIEffect.cpp

3.41 PySIStartup Class Reference

Inheritance diagram for PySIStartup:



Collaboration diagram for PySIStartup:



Static Public Member Functions

- static bp::tuple **context_dimensions** ()
- static void **create_region_by_type** (const bp::list &shape, int effect_type, bp::dict &kwargs)
- static void **create_region_by_name** (const bp::list &contour, const std::string &name, bp::dict &kwargs)
- static void **create_region_by_class** (const bp::list &contour, bp::object &clazz, bp::dict &kwargs)
- static void **logger_quench_messages_from_class** (const std::string &class_name)
- static void **logger_unquench_messages_from_class** (const std::string &class_name)
- static void **logger_log** (bool flag)
- static void **logger_set_log_output** (int32_t flags)
- static void **set_tangible_ip_address_and_port** (const std::string &ip, int port)
- static void **set_pen_color** (int color_id)
- static void **enable** (int32_t flags)
- static void **disable** (int32_t flags)
- static void **set_file_system_root_folder** (const std::string &path)
- static void **set_file_system_desktop_folder** (const std::string &path)
- static std::string **file_system_root_folder** ()
- static std::string **file_system_desktop_folder** ()
- static void **exclude_plugins** (const bp::list &plugins)

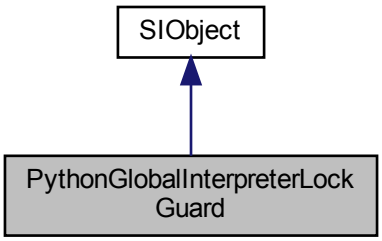
Additional Inherited Members

The documentation for this class was generated from the following files:

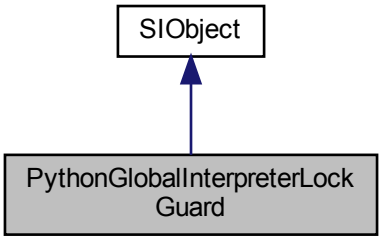
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/PySIStartup.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/PySIStartup.cpp

3.42 PythonGlobalInterpreterLockGuard Class Reference

Inheritance diagram for PythonGlobalInterpreterLockGuard:



Collaboration diagram for PythonGlobalInterpreterLockGuard:



Public Member Functions

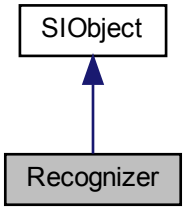
- **PythonGlobalInterpreterLockGuard** (const PythonGlobalInterpreterLockGuard &)=delete
- PythonGlobalInterpreterLockGuard & **operator=** (const PythonGlobalInterpreterLockGuard &)=delete

The documentation for this class was generated from the following file:

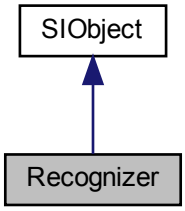
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/plugin/PythonGlobalInterpreterLock↩ Guard.hpp

3.43 Recognizer Class Reference

Inheritance diagram for Recognizer:



Collaboration diagram for Recognizer:



Public Member Functions

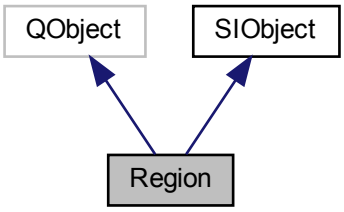
- float **recognize** (std::vector< glm::vec3 > &out, const std::vector< glm::vec3 > &in)
- const std::vector< Template > & **templates** () const

The documentation for this class was generated from the following files:

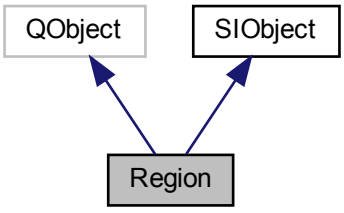
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/Dollar1GestureRecognizer.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/Dollar1GestureRecognizer.cpp

3.44 Region Class Reference

Inheritance diagram for Region:



Collaboration diagram for Region:



Public Member Functions

- **Region** (const std::vector< glm::vec3 > &contour, const bp::object &effect, uint32_t width=0, uint32_t height=0, bp::dict kwargs=bp::dict())
- **Region** (const bp::object &o, const bp::dict &qml, uint32_t width=0, uint32_t height=0)
- bool **is_transformed** () const
- void **set_is_transformed** (bool b)
- const std::string & **uuid** () const
- void **set_effect** (const bp::object &effect, bp::dict &kwargs)
- void **set_effect** (const std::vector< glm::vec3 > &contour, const bp::object &effect, const std::string &uuid, bp::dict &kwargs)
- void **set_data** (const QMap< QString, QVariant > &data)
- PySIEffect * **effect** ()
- bp::object & **raw_effect** ()
- const std::unique_ptr< RegionMask > & **mask** () const
- const std::vector< glm::vec3 > & **aabb** ()
- const std::vector< glm::vec3 > & **contour** ()
- const std::string & **qml_path** () const

- void **move_and_rotate** ()
- const glm::mat3x3 & **transform** () const
- uint8_t **on_enter** (PySIEffect *other)
- uint8_t **on_continuous** (PySIEffect *other)
- uint8_t **on_leave** (PySIEffect *other)
- Q_SIGNAL void **LINK_SIGNAL** (const std::string &uuid_event, const std::string &uuid_sender, const std::string &source_cap, const bp::object &args)
- Q_SLOT void **LINK_SLOT** (const std::string &uuid_event, const std::string &uuid_sender, const std::string &source_cap, const bp::object &args)
- Q_SLOT void **REGION_DATA_CHANGED_SLOT** (const QMap< QString, QVariant > &data)
- void **register_link_event** (const std::string &uuid, const std::string &attribute)
- void **register_link_event** (const std::tuple< std::string, std::string > &link_event)
- bool **is_link_event_registered** (const std::string &uuid, const std::string &attribute)
- bool **is_link_event_registered** (const std::tuple< std::string, std::string > &link_event)
- const std::string & **name** () const
- const glm::vec4 & **color** () const
- const uint16_t **type** () const
- const uint32_t **width** () const
- const uint32_t **height** () const
- const uint32_t **visualization_width** () const
- const uint32_t **visualization_height** () const
- uint8_t **handle_collision_event** (const std::string &function_name, PySIEffect *colliding_effect)
- void **update** ()
- const QMap< QString, QVariant > & **data** () const
- const int32_t **last_delta_x** () const
- const int32_t **last_delta_y** () const
- bool **is_new** ()
- void **set_is_new** (bool toggle)
- int32_t **x** ()
- int32_t **y** ()
- std::vector< int > & **grid_nodes** ()
- glm::ivec4 & **grid_bounds** ()
- float **angle** ()

Data Fields

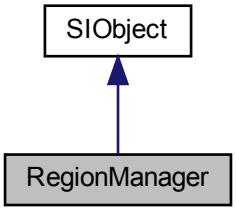
- int32_t **d_last_delta_x**
- int32_t **d_last_delta_y**
- int32_t **d_last_x**
- int32_t **d_last_y**

The documentation for this class was generated from the following files:

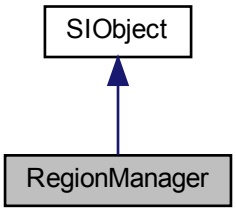
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/region/Region.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/region/Region.cpp

3.45 RegionManager Class Reference

Inheritance diagram for RegionManager:



Collaboration diagram for RegionManager:



Public Member Functions

- void **add_region** (const std::vector< glm::vec3 > &contour, const bp::object &effect, const bp::dict &kwargs=bp::dict())
- void **add_region** (const bp::object &o, const bp::dict &qml)
- std::vector< std::shared_ptr< Region > > & **regions** ()
- std::unordered_map< std::string, std::vector< glm::vec3 > > & **partial_regions** ()
- void **set_partial_regions** (const std::unordered_map< std::string, std::vector< glm::vec3 >> &partials)
- void **update** ()

Friends

- class **SIGRunRegionManagerTest**

The documentation for this class was generated from the following files:

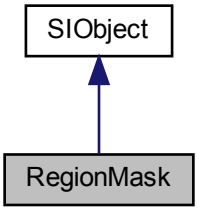
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/RegionManager.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/RegionManager.cpp

3.46 RegionMask Class Reference

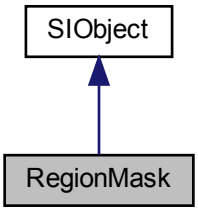
RegionMask class which stores a bit array used for true collision testing.

`#include <RegionMask.hpp>`

Inheritance diagram for RegionMask:



Collaboration diagram for RegionMask:



Public Member Functions

- RegionMask (uint32_t canvas_width, uint32_t canvas_height, const std::vector< glm::vec3 > &contour)
  constructor of the RegionMask class
- RegionMask (const RegionMask &rm)
  copy constructor
- ~RegionMask ()
  default destructor
- uint32_t size () const
  retrieve the size of the mask datastructure
- void set_bit (int32_t i)
  set the bit at index i of d_values to one/true
- void set_bit (const glm::vec3 &v)

set the bit at point v to one/true in d_values

- void clear_bit (int32_t i)

  set the bit at index i of d_values to zero/false
- void clear_bit (const glm::vec3 &v)

  set the bit at point v of d_values to zero/false
- void move (const glm::vec2 &v)

  update the AABB relations according to desired translation of a parent Region
- bool operator[ ] (int32_t i) const

  [] operator overloaded for returning the value of d_values at index i
- bool operator[ ] (const glm::vec3 &v) const

  [] operator overloaded for returning the value of d_values at point v

Friends

- class **SIGRunRegionMaskTest**

3.46.1 Detailed Description

RegionMask class which stores a bit array used for true collision testing.

Functionality
RegionMask class storing a bit array as std::vector<bool>. std::vector<bool> has a special implementation where its bool is stored in exactly one bit. See: https://en.cppreference.com/w/cpp/container/vector_bool This vector has the size of width * height of the AABB of the contour of the parent region. The array is filled with ones and zeroes according to a scanline algorithm. Every pixel which is part of the parent Region is set to one in that way. Others are left at 0. The array is relatively accessed according to the top left corner of that AABB.
Rationale:
The use of the AABB allows for creating a secondary coordinate system which is translated relatively to the parent coordinate system (canvas coordinate system). Therefore, each point which is to be tested with the mask is subtracted by the position vector of the AABB. In this way, that point in the canvas coordinate system is converted to the mask coordinate system. This leads to querying collosion occurrences relatively to the AABB. Through that, simple region translation does not require recomputation of the mask. Instead, the internal AABB is translated the same amount and the coordinate system conversion provides correct collision detection behaviour.

3.46.2 Constructor & Destructor Documentation

3.46.2.1 RegionMask() [1/2]

```
RegionMask::RegionMask (
            uint32_t canvas_width,
            uint32_t canvas_height,
            const std::vector< glm::vec3 > & contour )
```

constructor of the RegionMask class

Constructor of the RegionMask class. Initializes all datastructures required to maintain a RegionMask for Collision↩ Detection according to parameters. Performs scanline algorithm for generation of the actual mask relatively to AABB of parent region.

Parameters

canvas_width	int containing the width of the canvas
canvas_height	int containing the height of the canvas
contour	constant reference to a std::vector object containing glm::vec3 objects containing all the points of the contour of the parent Region
aabb	constant reference to a std::vector object containing glm::vec3 objects containing the four points of the AABB of the parent Region

See also

d_canvas_width
d_canvas_height
d_tlc_aabb_x
d_tlc_aabb_y
d_brc_aabb_x
d_brc_aabb_y
d_width_aabb
d_height_aabb
d_values

3.46.2.2 RegionMask() [2/2]

```
RegionMask::RegionMask (
            const RegionMask & rm )
```

copy constructor

Parameters

rm	the constant reference to a RegionMask object to be copied

See also

d_canvas_width
d_canvas_height
d_tlc_aabb_x
d_tlc_aabb_y
d_brc_aabb_x
d_brc_aabb_y
d_width_aabb
d_height_aabb
d_values

3.46.3 Member Function Documentation

3.46.3.1 clear_bit() [1/2]

```
void RegionMask::clear_bit (
            int32_t i )
```

set the bit at index i of d_values to zero/false

Sets the bit at index i of d_values to zero or false according to a bounds check.

Parameters

i	int which contains the index of the bit to be set to zero/false in d_values

See also

d_values

3.46.3.2 clear_bit() [2/2]

```
void RegionMask::clear_bit (
            const glm::vec3 & v )
```

set the bit at point v of d_values to zero/false

Sets the bit at point v of d_values to zero or false according to a bounds check. The bounds check is performed based on the actual index of the bit to set. The actual index is calculated according to AABB_WIDTH $*$ (v.y - A↩ ABB_TOP_LEFT_CORNER_Y) + v.x - AABB_TOP_LEFT_CORNER_X If the bounds check is negative, nothing happens.

Parameters

v	a constant reference to a glm::vec3 object containing the corresponding coordinates of the point to a bit of d_values which is to be set to zero or false.

See also

d_values
d_width_aabb
d_tlc_aabb_y
d_tlc_aabb_x

3.46.3.3 move()

```
void RegionMask::move (
            const glm::vec2 & v )
```

update the AABB relations according to desired translation of a parent Region

Use of RegionMasks occurs relatively to the AABB of its parent Region. Is the parent region moved / translated within the canvas, the AABB is also moved or translated. Therefore, the RegionMasks is upated according to that translation by storing the new translation parameters. In this way, the mask coordinate system is moved within the canvas coordinate system. So, after updating the AABB with the new translation values, the mask continues to function, due to its relative dependence on the AABB. Therefore, no recomputation is required.

Parameters

v	a constant reference to a glm::vec2 object containing the translation vector

3.46.3.4 operator[]() [1/2]

```
bool RegionMask::operator[] (
            int32_t i ) const
```

[] operator overloaded for returning the value of d_values at index i

Overloads the [] operator. Retrieves the bool value at index i of d_values.

Parameters

i	int containing the index

Returns

a bool containing whether the queried bit is set or not in d_values

See also

d_values

3.46.3.5 operator[]() [2/2]

```
bool RegionMask::operator[] (
            const glm::vec3 & v ) const
```

[] operator overloaded for returning the value of d_values at point v

Overloads the [] operator. Retrieves the bool value at point v of d_values. The actual index is calculated according to AABB_WIDTH $*$ (v.y - AABB_TOP_LEFT_CORNER_Y) + v.x - AABB_TOP_LEFT_CORNER_X.

Parameters

v	a constant reference to a glm::vec3 object containing the corresponding coordinates of the point to a bit of d_values which is to be tested whether the queried bit is set or not in d_values.

Returns

a bool containing whether the queried bit is set or not in d_values

See also

d_values
d_width_aabb
d_tlc_aabb_x
d_tlc_aabb_y

3.46.3.6 set_bit() [1/2]

```
void RegionMask::set_bit (
            int32_t i )
```

set the bit at index i of d_values to one/true

Sets the bit at index i of d_values to one or true according to a bounds check.

Parameters

i	int which contains the index of the bit to be set to one/true in d_values

See also

d_values

3.46.3.7 set_bit() [2/2]

```
void RegionMask::set_bit (
            const glm::vec3 & v )
```

set the bit at point v to one/true in d_values

Sets the bit at point v of d_values to one or true according to a bounds check. The bounds check is performed based on the actual index of the bit to be set. The actual index is calculated according to AABB_WIDTH $*$ (v.y - AABB_TOP_LEFT_CORNER_Y) + v.x - AABB_TOP_LEFT_CORNER_X If the bounds check is negative, the bit will be set to false.

Parameters

v	a constant reference to a glm::vec3 object containing the corresponding coordinates of the point to a bit of d_values which is to be set to one or true.

See also

d_values
d_width_aabb
d_tlc_aabb_y
d_tlc_aabb_x

3.46.3.8 size()

```
uint32_t RegionMask::size ( ) const
```

retrieve the size of the mask datastructure

Retrieves the size of the mask datastructure. This datastructure is called d_values.

Returns

the size of d_values

See also

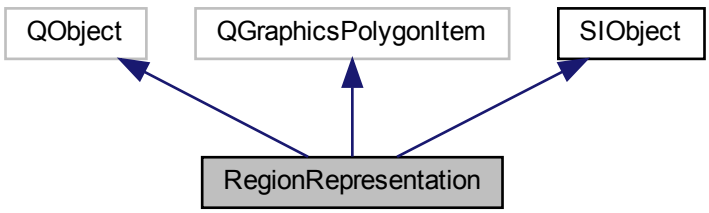
d_values

The documentation for this class was generated from the following files:

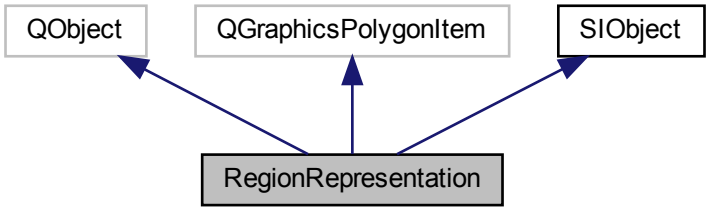
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/region/RegionMask.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/region/RegionMask.cpp

3.47 RegionRepresentation Class Reference

Inheritance diagram for RegionRepresentation:



Collaboration diagram for RegionRepresentation:



Public Member Functions

- **RegionRepresentation** (QQmlContext *c, const std::shared_ptr< Region > &region, QGraphicsView *parent)
- void **update** (const std::shared_ptr< Region > &region)
- const std::string & **uuid** () const
- const std::string & **name** () const
- const std::string & **qml_path** () const
- QColor & **color** ()
- void **paint** (QPainter *painter, const QStyleOptionGraphicsItem *option, QWidget *widget) override
- Q_SLOT void **set_data** (const QVariantMap &data)

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/region/RegionRepresentation.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/region/RegionRepresentation.cpp

3.48 RegionResampler Class Reference

Static Public Member Functions

- static void **resample** (std::vector< glm::vec3 > &out, const std::vector< glm::vec3 > &in, int step_count=SI_CONTOUR_STEPCOUNT)

Friends

- class **SIGRunRegionResamplerTest**

The documentation for this class was generated from the following files:

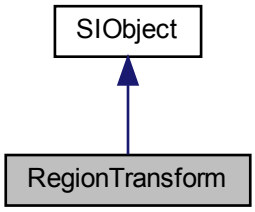
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/region/RegionResampler.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/region/RegionResampler.cpp

3.49 RegionTransform Class Reference

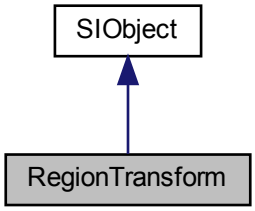
RegionTransform class storing the relative translation, rotation and scale of a contour.

`#include <RegionTransform.hpp>`

Inheritance diagram for RegionTransform:



Collaboration diagram for RegionTransform:



Public Member Functions

- RegionTransform ()

  default constructor initializing instance variables to default values
- ~RegionTransform ()

  default destructor
- void update (const glm::vec2 &translation=glm::vec2(0, 0), float angle=0.0, float scale=1.0, const glm::vec2 &rotation_origin=glm::vec2(0, 0))

  central function to update transformation matrix with new, relative translation, relative rotation and absolute scale values
- const glm::mat3x3 & transform ()
- const glm::vec3 & operator[ ] (uint32_t index)

  overloading of [] operator
- const glm::vec3 **operator** ∗ (const glm::vec3 &p)
- const glm::mat3x3 **mult** (const glm::mat3x3 &n, const glm::mat3x3 &m)

3.49.1 Detailed Description

RegionTransform class storing the relative translation, rotation and scale of a contour.

This class stores the relative translation, rotation and scale of a contour. The initial contour remains unchanged and change in one of those three aspects does mutate this transform but not the initial contour. The translation, rotation and scale are stored as a 3x3 transformation matrix. The transformation matrix is stored ROW MAJOR and requires LEFT or PRE-Multiplication. Therefore, multiplications with points look such as : p $*$ T, where p is a point and T is the transformation matrix. Due to matrix multiplications being not commutative, T $*$ p will not yield desired results.

See also

d_translation
d_rotation
d_scale
d_transform
d_angle

3.49.2 Constructor & Destructor Documentation

3.49.2.1 RegionTransform()

```
RegionTransform::RegionTransform ( )
```

default constructor initializing instance variables to default values

Default constructor. Initializes all matrix objects to identity matrices. Sets cumulative angle to 0

See also

d_transform
d_translation
d_rotation
d_scale \scale d_angle

3.49.2.2 ~RegionTransform()

```
RegionTransform::~RegionTransform ( )
```

default destructor

Default destructor.

3.49.3 Member Function Documentation

3.49.3.1 operator[]()

```
const glm::vec3 & RegionTransform::operator[] (
            uint32_t index )
```

overloading of [] operator

Overloading of [] operator. Makes it easier to use the transformation matrix stored in this class. This function returns a constant glm::vec3 reference which itself is subscriptable with the [] operator.

Parameters

index	an integer containing the index of the row of the transformation matrix to be retrieved.

Returns

a constant reference of glm::vec3 object containing the queried row of the transformation matrix

See also

d_transform

3.49.3.2 transform()

```
const glm::mat3x3 & RegionTransform::transform ( )
```

Returns

a constant reference to a glm::mat3x3 object containing the current transformation matrix

See also

d_transform

3.49.3.3 update()

```
void RegionTransform::update (
            const glm::vec2 & translation = glm::vec2(0, 0),
            float angle = 0.0,
            float scale = 1.0,
            const glm::vec2 & rotation_origin = glm::vec2(0, 0) )
```

central function to update transformation matrix with new, relative translation, relative rotation and absolute scale values

Updates translation matrix T, rotation matrix R, and scale matrix S according to the given parameters. Too small angle increments are ignored to save computations of required trigonometric functions. Computes a the new transformation matrix according to $T * R * S$.

Parameters

translation	a constant reference to a glm::vec2 datastructure containing the new, relative translation of the parent contour
angle	a float containing the new relative angle of the parent contour according to x-axis
scale	a float containing the new absolute scale factor of the contour

See also

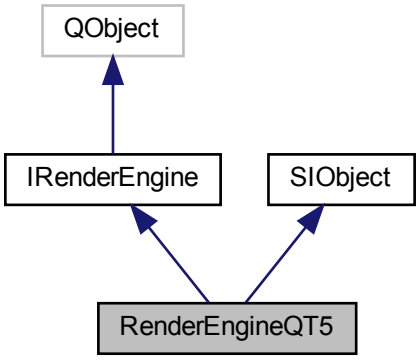
d_translation
d_angle
d_rotation
d_scale
d_transform

The documentation for this class was generated from the following files:

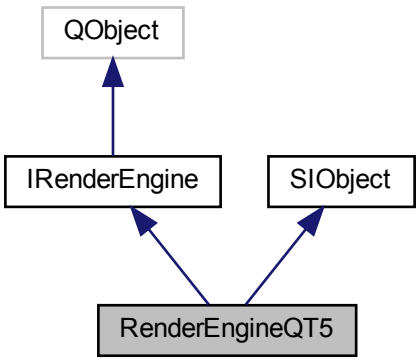
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/region/RegionTransform.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/region/RegionTransform.cpp

3.50 RenderEngineQT5 Class Reference

Inheritance diagram for RenderEngineQT5:



Collaboration diagram for RenderEngineQT5:



Public Member Functions

- void **start** (uint32_t width, uint32_t height, uint32_t target_fps=60) override
- void **run** () override
- void **pause** () override
- void **stop** () override
- void **disable_anti_aliasing** () override
- void **enable_anti_aliasing** (uint32_t samplng_factor) override
- void **set_cursor_stroke_width_by_cursor_id** (const std::string &cursor_id, int stroke_width) override
- void **set_cursor_stroke_color_by_cursor_id** (const std::string &cursor_id, const glm::vec4 &color) override

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/RenderEngineQt5.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/RenderEngineQt5.cpp

3.51 Result Class Reference

Public Member Functions

- **Result** (const std::string &name, float score)
- const std::string & **name** () const
- float **score** () const

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/Dollar1GestureRecognizer.hpp

3.52 RingBuffer< T > Class Template Reference

Public Member Functions

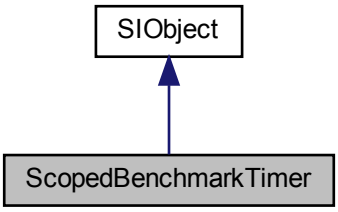
- **RingBuffer** (int size)
- void **push_back** (const T &data)
- const T & **get** ()
- bool **find** (const T &data) const
- void **clear** ()
- bool **empty** () const
- uint32_t **size** () const
- uint32_t **max_size** () const
- bool **operator &** (const T &value) const
- void **operator<<** (const T &value)
- const std::vector< T > & **buffer** () const

The documentation for this class was generated from the following file:

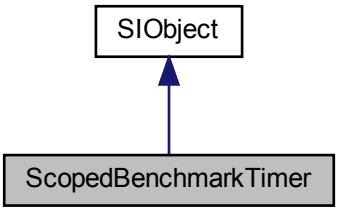
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/RingBuffer.hpp

3.53 ScopedBenchmarkTimer Class Reference

Inheritance diagram for ScopedBenchmarkTimer:



Collaboration diagram for ScopedBenchmarkTimer:



Public Member Functions

- long **stop** ()

Data Fields

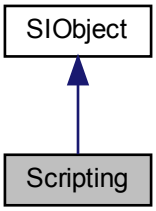
- std::chrono::time_point< std::chrono::high_resolution_clock > **d_start_point**

The documentation for this class was generated from the following file:

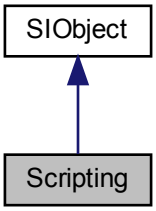
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/Benchmark.hpp

3.54 Scripting Class Reference

Inheritance diagram for Scripting:



Collaboration diagram for Scripting:



Public Member Functions

- std::string **transpile** (std::string &path, const std::string &path_addition)
- bp::object **si_plugin** (std::string &module_name, std::string &path)

Friends

- std::ostream & **operator**<< (std::ostream &os, const Scripting &scripting)

The documentation for this class was generated from the following files:

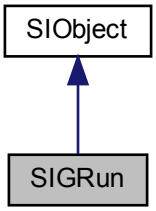
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/plugin/Scripting.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/plugin/Scripting.cpp

3.55 SIGRun Class Reference

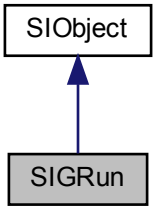
SIGRun class serving as entry point of an SI environment.

`#include <SIGRun.hpp>`

Inheritance diagram for SIGRun:



Collaboration diagram for SIGRun:



Public Member Functions

- SIGRun ()
  constructor
- ∼SIGRun ()
  destructor
- int exec (int argc, char ∗∗argv, IRenderEngine ∗ire, IPhysicalEnvironment ∗ros)
  entry point of SIGRun

Static Public Member Functions

- static int quit ()
  exit SIGRun

3.55.1 Detailed Description

SIGRun class serving as entry point of an SI environment.

This class serves as the entry point of an SI environment. It is directly exposed in SI.hpp. An instance of this class is used to launch an SI environment.

See also

up_core

3.55.2 Constructor & Destructor Documentation

3.55.2.1 SIGRun()

```
SIGRun::SIGRun ( )
```

constructor

Constructor of SIGRun class. Used for instantiating objects.

3.55.2.2 ~SIGRun()

```
SIGRun::~SIGRun ( )
```

destructor

Destructor of SIGRun class. Used for destroying objects.

3.55.3 Member Function Documentation

3.55.3.1 exec()

```
int SIGRun::exec (
            int argc,
            char ** argv,
            IRenderEngine * ire,
            IPhysicalEnvironment * ros )
```

entry point of SIGRun

Entry point of SIGRun initializing all further systems.

Parameters

argc	cli argc
argv	cli argv

3.55.3.2 quit()

```
int SIGRun::quit ( )  [static]
```

exit SIGRun

static exit function of SIGRun terminating all other systems

The documentation for this class was generated from the following files:

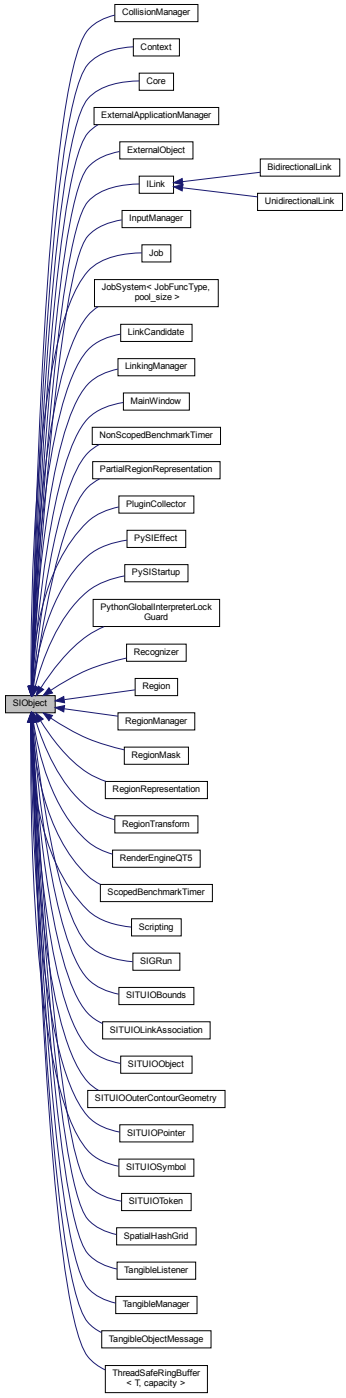
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/SIGRun.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/SIGRun.cpp

3.56 SIObject Class Reference

A meta class from which other classes are derived from to register them as SIObject meta types.

```
#include <SIObject.hpp>
```

Inheritance diagram for SIObject:



Public Member Functions

- virtual const std::string & meta_type () const =0

  function for retrieving meta type name
- virtual const std::string & origin () const =0

  function for retrieving origin

3.56.1 Detailed Description

A meta class from which other classes are derived from to register them as SIObject meta types.

This class enables registering other classes as SIObject meta types. This is currently achieved by storing std::strings containing the classes individual names. Currently, this meta typing is only used for Logging.

See also

Log::Log
d_meta_type

3.56.2 Member Function Documentation

3.56.2.1 meta_type()

```
virtual const std::string& SIObject::meta_type ( ) const  [pure virtual]
```

function for retrieving meta type name

The function for retrieving meta type name in a constant manner. Therefore, the instance calling this function will not mutate.

Returns

d_meta_type a const std::string reference of the type name of the class

3.56.2.2 origin()

```
virtual const std::string& SIObject::origin ( ) const  [pure virtual]
```

function for retrieving origin

The function for retrieving origin in a constant manner. Therefore, the instance calling this function will not mutate.

Returns

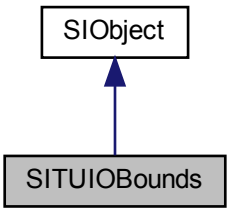
origin a const std::string reference of the origin of the class

The documentation for this class was generated from the following file:

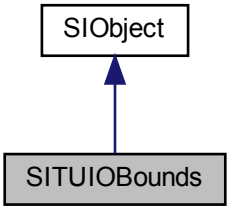
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/SIObject.hpp

3.57 SITUIOBounds Class Reference

Inheritance diagram for SITUIOBounds:



Collaboration diagram for SITUIOBounds:



Public Member Functions

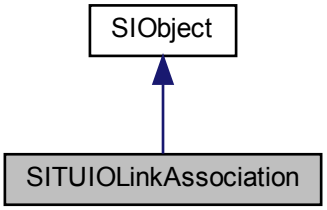
- **SITUIOBounds** (const osc::ReceivedMessage &m)
- int **s_id** ()
- float **x_pos** () const
- float **y_pos** ()
- float **angle** ()
- float **width** ()
- float **height** ()
- float **area** ()
- float **x_vel** ()
- float **y_vel** ()
- float **a_vel** ()
- float **m_acc** ()
- float **r_acc** ()

The documentation for this class was generated from the following files:

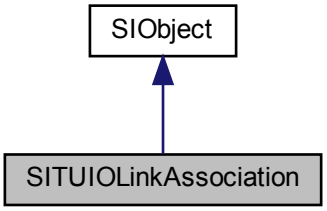
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.cpp

3.58 SITUIOLinkAssociation Class Reference

Inheritance diagram for SITUIOLinkAssociation:



Collaboration diagram for SITUIOLinkAssociation:



Public Member Functions

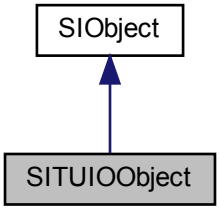
- **SITUIOLinkAssociation** (const std::vector< int > &link_associations)
- const std::vector< int > & **link_associations** ()

The documentation for this class was generated from the following files:

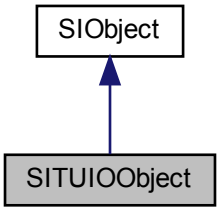
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.cpp

3.59 SITUIOObject Class Reference

Inheritance diagram for SITUIOObject:



Collaboration diagram for SITUIOObject:



Public Member Functions

- **SITUIOObject** (int s_id, int f_id, int source_width, int source_height)
- void **add_token_data** (const osc::ReceivedMessage &m)
- void **add_pointer_data** (const osc::ReceivedMessage &m)
- void **add_bounds_data** (const osc::ReceivedMessage &m)
- void **add_symbol_data** (const osc::ReceivedMessage &m)
- void **add_outer_contour_geometry_data** (const osc::ReceivedMessage &m)
- void **add_link_association_data** (const std::vector< int > &link_associations)
- SITUIOToken *const **token_component** () const
- SITUIOPointer *const **pointer_component** () const
- SITUIOBounds *const **bounds_component** () const
- SITUIOSymbol *const **symbol_component** () const
- const SITUIOOuterContourGeometry * **outer_contour_geometry_component** () const
- SITUIOLinkAssociation *const **link_association** () const
- bool **has_token_component** ()
- bool **has_pointer_component** ()
- bool **has_bounds_component** ()

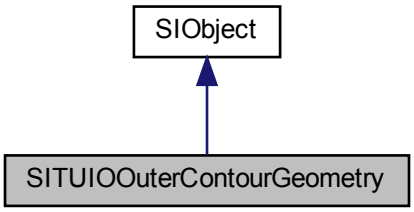
- bool **has_symbol_component** ()
- bool **has_outer_counter_geometry_component** ()
- bool **has_linking_association_component** ()
- bool **has_any_component** ()
- int **s_id** ()
- int **source_width** ()
- int **source_height** ()

The documentation for this class was generated from the following files:

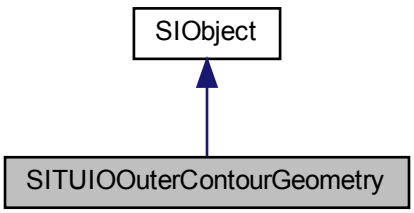
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.cpp

3.60 SITUIOOuterContourGeometry Class Reference

Inheritance diagram for SITUIOOuterContourGeometry:



Collaboration diagram for SITUIOOuterContourGeometry:



Public Member Functions

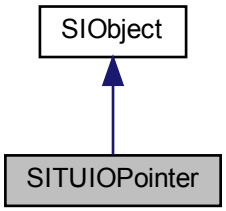
- **SITUIOOuterContourGeometry** (const osc::ReceivedMessage &m)
- int **s_id** ()
- const std::vector< glm::vec3 > & **contour** () const

The documentation for this class was generated from the following files:

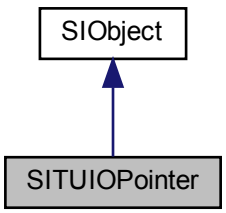
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.cpp

3.61 SITUIOPointer Class Reference

Inheritance diagram for SITUIOPointer:



Collaboration diagram for SITUIOPointer:



Public Member Functions

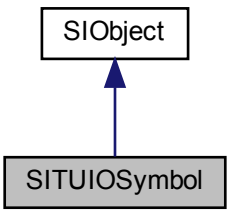
- **SITUIOPointer** (const osc::ReceivedMessage &m)
- int **s_id** ()
- int **t_id** ()
- int **u_id** ()
- int **c_id** ()
- float **x_pos** ()
- float **y_pos** ()
- float **angle** ()
- float **shear** ()
- float **radius** ()
- float **press** ()
- float **x_vel** ()
- float **y_vel** ()
- float **p_vel** ()
- float **m_acc** ()
- float **r_acc** ()

The documentation for this class was generated from the following files:

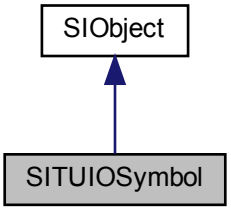
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.cpp

3.62 SITUIOSymbol Class Reference

Inheritance diagram for SITUIOSymbol:



Collaboration diagram for SITUIOSymbol:



Public Member Functions

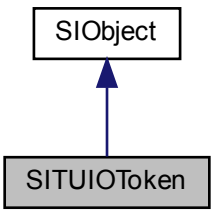
- **SITUIOSymbol** (const osc::ReceivedMessage &m)
- int **s_id** ()
- int **t_id** ()
- int **u_id** ()
- int **c_id** ()
- const std::string & **group** () const
- const std::string & **data** () const

The documentation for this class was generated from the following files:

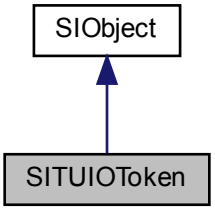
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.cpp

3.63 SITUIOToken Class Reference

Inheritance diagram for SITUIOToken:



Collaboration diagram for SITUIOToken:



Public Member Functions

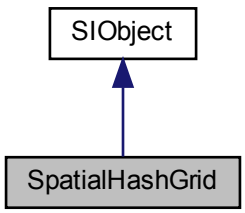
- **SITUIOToken** (const osc::ReceivedMessage &m)
- int **s_id** ()
- int **t_id** ()
- int **u_id** ()
- int **c_id** ()
- float **x_pos** ()
- float **y_pos** ()
- float **angle** () const
- float **x_vel** ()
- float **y_vel** ()
- float **a_vel** ()
- float **m_acc** ()
- float **r_acc** ()

The documentation for this class was generated from the following files:

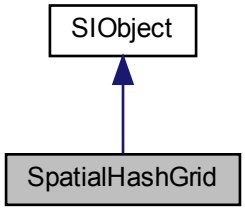
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/SITUIOObject.cpp

3.64 SpatialHashGrid Class Reference

Inheritance diagram for SpatialHashGrid:



Collaboration diagram for SpatialHashGrid:



Public Member Functions

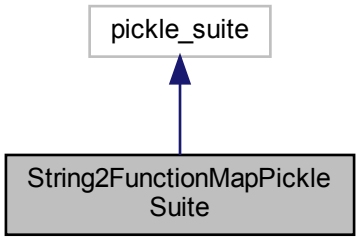
- **SpatialHashGrid** (int width, int height, int cells_per_row, int cells_per_column)
- void **update_region** (Region ∗r)
- void **register_region** (Region ∗r)
- bool **has_shared_cell** (Region ∗a, Region ∗b)

The documentation for this class was generated from the following files:

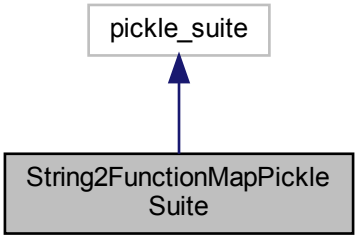
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/spatial_grid/SpatialHashGrid.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/spatial_grid/SpatialHashGrid.cpp

3.65 String2FunctionMapPickleSuite Class Reference

Inheritance diagram for String2FunctionMapPickleSuite:



Collaboration diagram for String2FunctionMapPickleSuite:



Static Public Member Functions

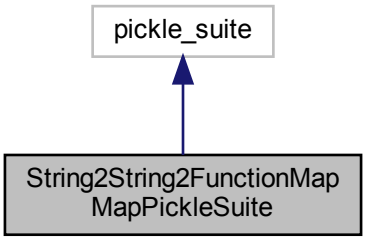
- static bp::tuple **getinitargs** (std::unordered_map< std::string, bp::object > m)

The documentation for this class was generated from the following file:

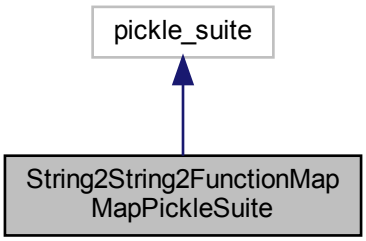
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.66 String2String2FunctionMapMapPickleSuite Class Reference

Inheritance diagram for String2String2FunctionMapMapPickleSuite:



Collaboration diagram for String2String2FunctionMapMapPickleSuite:



Static Public Member Functions

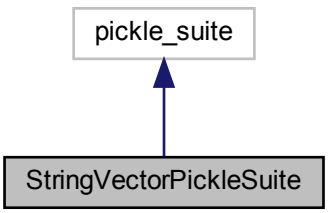
- static bp::tuple **getinitargs** (std::unordered_map< std::string, std::unordered_map< std::string, bp::object >> m)

The documentation for this class was generated from the following file:

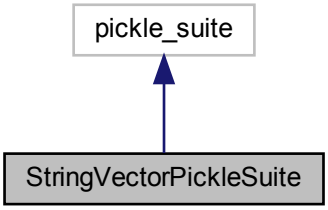
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.67 StringVectorPickleSuite Class Reference

Inheritance diagram for StringVectorPickleSuite:



Collaboration diagram for StringVectorPickleSuite:



Static Public Member Functions

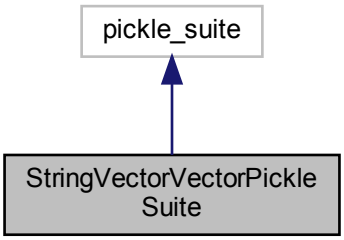
- static bp::tuple **getinitargs** (std::vector< std::string > &v)

The documentation for this class was generated from the following file:

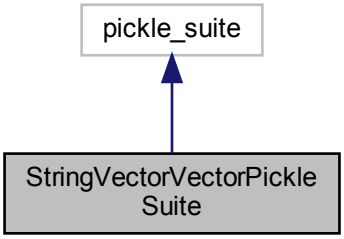
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.68 StringVectorVectorPickleSuite Class Reference

Inheritance diagram for StringVectorVectorPickleSuite:



Collaboration diagram for StringVectorVectorPickleSuite:



Static Public Member Functions

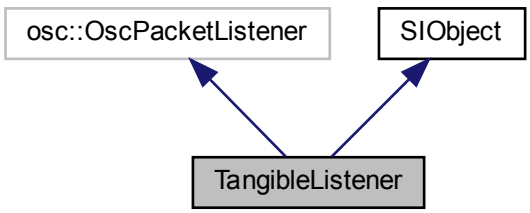
- static bp::tuple **getinitargs** (std::vector< std::vector< std::string >> &vs)

The documentation for this class was generated from the following file:

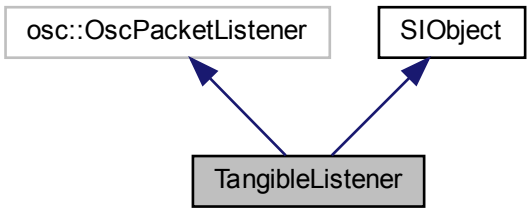
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/pickling/PickleSuits.hpp

3.69 TangibleListener Class Reference

Inheritance diagram for TangibleListener:



Collaboration diagram for TangibleListener:



Protected Member Functions

- void **ProcessMessage** (const osc::ReceivedMessage &m, const IpEndpointName &remoteEndpoint) override

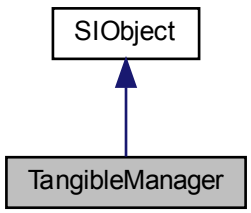
Additional Inherited Members

The documentation for this class was generated from the following files:

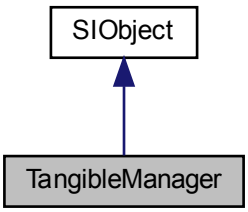
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/TangibleListener.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/_tangible/TangibleListener.cpp

3.70 TangibleManager Class Reference

Inheritance diagram for TangibleManager:



Collaboration diagram for TangibleManager:



Public Member Functions

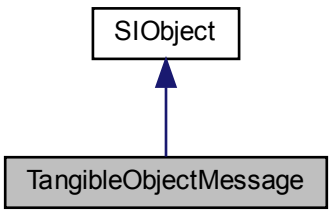
- void **start** ()
- void ∗ **handle_uds** (void ∗args)
- bool **is_started** ()

The documentation for this class was generated from the following files:

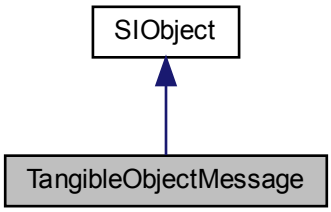
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/TangibleManager.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/TangibleManager.cpp

3.71 TangibleObjectMessage Class Reference

Inheritance diagram for TangibleObjectMessage:



Collaboration diagram for TangibleObjectMessage:



Public Member Functions

- **TangibleObjectMessage** (int32_t id, const std::vector< glm::vec3 > &shape, const std::string &plugin_↩ identifier, float x, float y, const glm::vec4 &color, bool is_click, bool is_drag, bool is_dbl_click, bool is_touch, bool is_alive, const std::vector< int > &links, int tracker_dimension_x, int tracker_dimension_y)
- void **send** ()
- const int **id** () const
- const std::vector< glm::vec3 > & **shape** () const
- const std::string & **plugin_identifier** () const
- const float **x** () const
- const float **y** () const
- const glm::vec4 & **color** () const
- const bool **is_click** () const
- const bool **is_drag** () const
- const bool **is_dbl_click** () const
- const bool **is_touch** () const
- const bool **is_alive** () const
- const std::vector< int > & **links** () const
- const glm::vec2 & **tracker_dimensions** () const
- const int **tracker_dimension_x** () const
- const int **tracker_dimension_y** () const
- const bool **has_links** () const

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/network/TangibleObjectMessage.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/network/TangibleObjectMessage.cpp

3.72 Template Class Reference

Public Member Functions

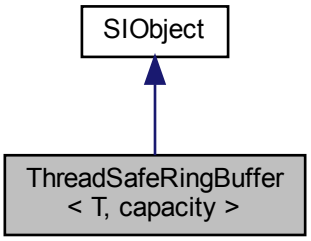
- **Template** (const std::string &name, const std::vector< glm::vec3 > &points)
- const std::vector< glm::vec3 > & **points** () const
- const std::string & **name** () const

The documentation for this class was generated from the following files:

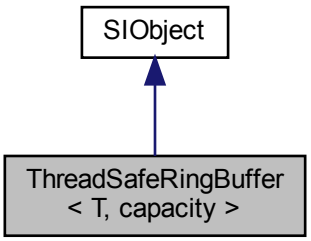
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/Dollar1GestureRecognizer.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/Dollar1GestureRecognizer.cpp

3.73 ThreadSafeRingBuffer< T, capacity > Class Template Reference

Inheritance diagram for ThreadSafeRingBuffer< T, capacity >:



Collaboration diagram for ThreadSafeRingBuffer< T, capacity >:



Public Member Functions

- bool **push_back** (const T &item)
- bool **pop_front** (T &item)

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/parallel/helpers/ThreadSafeRingBuffer.↩ hpp

3.74 Time Class Reference

Static Public Member Functions

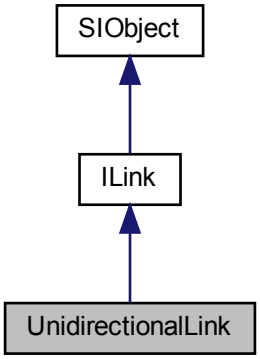
- static double **get_time** ()
- static void **set_time_delta** (double td)
- static double **time_delta** ()

The documentation for this class was generated from the following files:

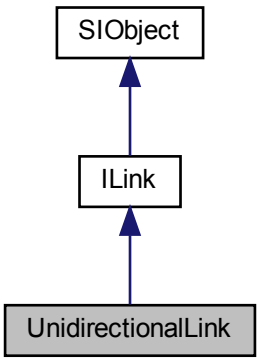
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/timing/Timing.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/siren/timing/Timing.cpp

3.75 UnidirectionalLink Class Reference

Inheritance diagram for UnidirectionalLink:



Collaboration diagram for UnidirectionalLink:



Public Member Functions

- **UnidirectionalLink** (const std::shared_ptr< Region > &ra, const std::shared_ptr< Region > &rb, const std::string &aa, const std::string &ab)
- **UnidirectionalLink** (const std::shared_ptr< ExternalObject > &eo, const std::shared_ptr< Region > &ra, const std::string &aa, const std::string &ab)
- const LINK_TYPE & **type** () const override
- const std::shared_ptr< Region > & **sender_a** () const override
- const std::shared_ptr< Region > & **sender_b** () const override
- const std::shared_ptr< Region > & **receiver_a** () const override
- const std::shared_ptr< Region > & **receiver_b** () const override
- const std::shared_ptr< ExternalObject > & **external_sender_a** () const override
- const std::string & **attribute_a** () const override
- const std::string & **attribute_b** () const override
- const bool **is_external** () const override
- virtual void **add_child** (std::shared_ptr< ILink > &link) override
- std::vector< std::shared_ptr< ILink > > & **children** () override

Additional Inherited Members

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/helpers/linking/Link.↩ hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/context/managers/helpers/linking/Link.↩ cpp

3.76 UUID Class Reference

Static Public Member Functions

- static std::string **uuid** ()

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/util/UUID.hpp

3.77 VectorExposure< T > Class Template Reference

VectorExposure class providing the interface for exposing STL vectors to the python3 bindings (PySI) in a pythonic way.

```
#include <VectorExposure.hpp>
```

Public Types

- typedef T::value_type V

Static Public Member Functions

- static V & get (T &self, uint32_t index)

  returns a reference to a value of type V at index i contained in STL vector T
- static T get_slice (T const &self, const bp::slice &i)

  enables pythonic slicing of exposed STL vector T with values of type V
- static void set (T &self, uint32_t index, V const &value)

  sets the value at the given index of self to the given value
- static void del (T &self, uint32_t index)

  removes the value at the given index, therefore reducing the vectors size by one
- static void add (T &self, V const &value)

  add a value of type V to the back of a STL vector of type T
- static constexpr bool in (T const &self, V const &value)

  check if a STL vector of type T with values of type V contains a value of type V
- static constexpr int index (T const &self, V const &value)

  returns the index of a value of type in the STL vector of type T with values of type V

3.77.1 Detailed Description

template<typename T>
class VectorExposure< T >

VectorExposure class providing the interface for exposing STL vectors to the python3 bindings (PySI) in a pythonic way.

Template Parameters

T	the STL vector to be exposed

3.77.2 Member Typedef Documentation

3.77.2.1 V

```
template<typename T >
typedef T::value_type VectorExposure< T >::V
```

Template Parameters

V	the type of variable the STL vector T contains

3.77.3 Member Function Documentation

3.77.3.1 add()

```
template<typename T >
static void VectorExposure< T >::add (
            T & self,
            V const & value )  [inline], [static]
```

add a value of type V to the back of a STL vector of type T

Parameters

in,out	*self*	the STL vector to receive the value
in	*value*	the value to be pushed to the back of self

3.77.3.2 del()

```
template<typename T >
static void VectorExposure< T >::del (
            T & self,
            uint32_t index )  [inline], [static]
```

removes the value at the given index, therefore reducing the vectors size by one

Parameters

in,out	*self*	the STL vector of type T with values of type V to have a value deleted
in	*index*	the index of the value to be deleted in self

3.77.3.3 get()

```
template<typename T >
static V& VectorExposure< T >::get (
            T & self,
            uint32_t index )  [inline], [static]
```

returns a reference to a value of type V at index i contained in STL vector T

Returns a reference to a value of type V at index i contained in STL vector T. Also allows pythonic accsess to values via negative indices

Parameters

in	*self*	the STL vector of type T with values of type V
in	*index*	the index of the value to be returned by reference

Returns

a reference to the value in self at the given index

3.77.3.4 get_slice()

```
template<typename T >
static T VectorExposure< T >::get_slice (
            T const & self,
            const bp::slice & i )  [inline], [static]
```

enables pythonic slicing of exposed STL vector T with values of type V

Parameters

in	*self*	the STL vector of type T and values of type V to be sliced
in	*i*	the slicing parameters

Returns

a deep copy to a STL vector of type T containing the elements of self which remained after slicing

3.77.3.5 in()

```
template<typename T >
static constexpr bool VectorExposure< T >::in (
            T const & self,
            V const & value )  [inline], [static]
```

check if a STL vector of type T with values of type V contains a value of type V

Parameters

in	*self*	the STL vector to be checked whether it contains the given value
in	*value*	the value to be checked whether it is contained in self

Returns

true if self contains the value and false else

3.77.3.6 index()

```
template<typename T >
static constexpr int VectorExposure< T >::index (
```

```
            T const & self,
            V const & value )  [inline], [static]
```

returns the index of a value of type in the STL vector of type T with values of type V

Return the index of a value in the STL vector or -1 if the value is not present in the vector

Parameters

in	*self*	the target vector
in	*value*	the value which index is to be returned

Returns

the index of the value in self or -1 if the value is not contained by self

3.77.3.7 set()

```
template<typename T >
static void VectorExposure< T >::set (
            T & self,
            uint32_t index,
            V const & value )  [inline], [static]
```

sets the value at the given index of self to the given value

Parameters

in,out	*self*	the vector of type T with values of type V which value at given index is to be changed
in	*index*	the index of the value to be changed
in	*value*	the new value to be set at the given index in self

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/VectorExposure.hpp

3.78 VectorExposureLinkRelation Class Reference

Special wrapper class for VectorExposure handling vectors of LinkRelation.

```
#include <VectorExposure.hpp>
```

Static Public Member Functions

- static boost::shared_ptr< std::vector< LinkCandidate > > init (const bp::list &list=bp::list())

  exposed constructor to create a vector of LinkRelation based on a python list
- static void add (std::vector< LinkCandidate > &self, const bp::list &list)

  adds a new LinkRelation to the given vector
- static void set (std::vector< LinkCandidate > &self, uint32_t index, const bp::list &list)

  sets the value at the given index of self to the given value
- static const std::string repr (std::vector< LinkCandidate > &self)

  *returns the vectors representation as a string (**repr** in python)*

3.78.1 Detailed Description

Special wrapper class for VectorExposure handling vectors of LinkRelation.

3.78.2 Member Function Documentation

3.78.2.1 add()

```
static void VectorExposureLinkRelation::add (
            std::vector< LinkCandidate > & self,
            const bp::list & list )  [inline], [static]
```

adds a new LinkRelation to the given vector

Parameters

in,out	*self*	a vector of LinkRelation to receive a new LinkRelation
in	*list*	a python list containing LinkRelation to be added to self

3.78.2.2 init()

```
static boost::shared_ptr<std::vector<LinkCandidate> > VectorExposureLinkRelation::init (
            const bp::list & list = bp::list() )  [inline], [static]
```

exposed constructor to create a vector of LinkRelation based on a python list

Parameters

in	*list*	a list containing instances of LinkRelation or one LinkRelation in form of four strings.

Returns

a reference to a new std::vector<LinkRelation>> exposed to python containing the values of list

See also

LinkRelation

3.78.2.3 repr()

```
static const std::string VectorExposureLinkRelation::repr (
            std::vector< LinkCandidate > & self )  [inline], [static]
```

returns the vectors representation as a string (**repr** in python)

Parameters

in	*self*	the vector which data is to be presented in a readable way

Returns

the std::string containing the vector's representation

3.78.2.4 set()

```
static void VectorExposureLinkRelation::set (
            std::vector< LinkCandidate > & self,
            uint32_t index,
            const bp::list & list )  [inline], [static]
```

sets the value at the given index of self to the given value

Parameters

in,out	*self*	the vector which value at given index is to be changed
in	*index*	the index of the value to be changed
in	*list*	the list containing LinkRelation to be applied at the given index

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/VectorExposure.hpp

3.79 VectorExposureString Class Reference

Special wrapper class for VectorExposure handling vectors of std::string.

`#include <VectorExposure.hpp>`

Static Public Member Functions

- static boost::shared_ptr< std::vector< std::string > > init (const bp::list &list=bp::list())
  exposed constructor to create a vector of std::string based on a python list
- static void add (std::vector< std::string > &self, const std::string &s)
  adds a new std::string to the given vector
- static void set (std::vector< std::string > &self, uint32_t index, const std::string &s)
  sets the value at the given index of self to the given value
- static const std::string repr (std::vector< std::string > &self)
  *returns the vectors representation as a string (**repr** in python)*

3.79.1 Detailed Description

Special wrapper class for VectorExposure handling vectors of std::string.

3.79.2 Member Function Documentation

3.79.2.1 add()

```
static void VectorExposureString::add (
            std::vector< std::string > & self,
            const std::string & s )  [inline], [static]
```

adds a new std::string to the given vector

Parameters

in,out	*self*	a vector of strings to receive a new string
in	*s*	a std::string to be added to self

3.79.2.2 init()

```
static boost::shared_ptr<std::vector<std::string> > VectorExposureString::init (
            const bp::list & list = bp::list() )  [inline], [static]
```

exposed constructor to create a vector of std::string based on a python list

Parameters

in	*list*	a list containing strings.

Returns

a reference to a new std::vector<std::string> exposed to python containing the values of list

3.79.2.3 repr()

```
static const std::string VectorExposureString::repr (
            std::vector< std::string > & self )  [inline], [static]
```

returns the vectors representation as a string (**repr** in python)

Parameters

in	*self*	the vector which data is to be presented in a readable way

Returns

the std::string containing the vector's representation

3.79.2.4 set()

```
static void VectorExposureString::set (
            std::vector< std::string > & self,
            uint32_t index,
            const std::string & s )  [inline], [static]
```

sets the value at the given index of self to the given value

Parameters

in,out	*self*	the vector which value at given index is to be changed
in	*index*	the index of the value in self to be changed
in	*s*	the std::string to be applied at the given index

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/VectorExposure.hpp

3.80 VectorExposureStringVector Class Reference

Special wrapper class for VectorExposureStringVector handling vectors of vectors of std::string.

`#include <VectorExposure.hpp>`

Static Public Member Functions

- static boost::shared_ptr< std::vector< std::vector< std::string > > > init (const bp::list &list=bp::list())

  exposed constructor to create a vector of vectors of std::string based on a python list
- static void add (std::vector< std::vector< std::string >> &self, const std::vector< std::string > &s)

  adds a new std::vector<std::string> to the given vector
- static void set (std::vector< std::vector< std::string >> &self, uint32_t index, const std::vector< std::string > &s)

  sets the value at the given index of self to the given value
- static const std::string repr (std::vector< std::vector< std::string >> &self)

  *returns the vectors representation as a string (**repr** in python)*

3.80.1 Detailed Description

Special wrapper class for VectorExposureStringVector handling vectors of vectors of std::string.

3.80.2 Member Function Documentation

3.80.2.1 add()

```
static void VectorExposureStringVector::add (
            std::vector< std::vector< std::string >> & self,
            const std::vector< std::string > & s )  [inline], [static]
```

adds a new std::vector<std::string> to the given vector

Parameters

in,out	*self*	a vector of vectors of strings to receive a new vector of strings
in	*s*	a std::vector<std::string> to be added to self

3.80.2.2 init()

```
static boost::shared_ptr<std::vector<std::vector<std::string> > > VectorExposureString↩
Vector::init (
            const bp::list & list = bp::list() )  [inline], [static]
```

exposed constructor to create a vector of vectors of std::string based on a python list

Parameters

in	*list*	a list of lists containing strings.

Returns

a reference to a new std::vector<std::vector<std::string>> exposed to python containing the values of list

3.80.2.3 repr()

```
static const std::string VectorExposureStringVector::repr (
            std::vector< std::vector< std::string >> & self )  [inline], [static]
```

returns the vectors representation as a string (**repr** in python)

Parameters

in	*self*	the vector which data is to be presented in a readable way

Returns

the std::string containing the vector's representation

3.80.2.4 set()

```
static void VectorExposureStringVector::set (
            std::vector< std::vector< std::string >> & self,
            uint32_t index,
            const std::vector< std::string > & s )  [inline], [static]
```

sets the value at the given index of self to the given value

Parameters

in,out	*self*	the vector which value at given index is to be changed
in	*index*	the index of the value in self to be changed
in	*s*	the std::vector<std::string> to be applied at the given index

The documentation for this class was generated from the following file:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/VectorExposure.hpp

3.81 VectorExposureVec3 Class Reference

Special wrapper class for VectorExposure handling vectors of glm::vec3.

`#include <VectorExposure.hpp>`

Static Public Member Functions

- static boost::shared_ptr< std::vector< glm::vec3 > > init (const bp::list &list=bp::list())
  exposed constructor to create a vector of glm::vec3 based on a python list
- static void add (std::vector< glm::vec3 > &self, const bp::list &list)
  adds a new point to the given vector
- static void set (std::vector< glm::vec3 > &self, uint32_t index, const bp::list &list)
  sets the value at the given index of self to the given value
- static const std::string repr (std::vector< glm::vec3 > &self)
  *returns the vectors representation as a string (**repr** in python)*

3.81.1 Detailed Description

Special wrapper class for VectorExposure handling vectors of glm::vec3.

3.81.2 Member Function Documentation

3.81.2.1 add()

```
static void VectorExposureVec3::add (
            std::vector< glm::vec3 > & self,
            const bp::list & list )  [inline], [static]
```

adds a new point to the given vector

Parameters

in,out	*self*	a vector of points to receive a new point
	list[in]	a python list containing a points coordinates to be added to self

3.81.2.2 init()

```
static boost::shared_ptr<std::vector<glm::vec3> > VectorExposureVec3::init (
            const bp::list & list = bp::list() )  [inline], [static]
```

exposed constructor to create a vector of glm::vec3 based on a python list

Parameters

in	*list*	a list containing either further lists of three floats each representing coordinates. Or containing three floats representing one point.

Returns

a reference to a new std::vector<glm::vec3> exposed to python containing the values of list

3.81.2.3 repr()

```
static const std::string VectorExposureVec3::repr (
            std::vector< glm::vec3 > & self )  [inline], [static]
```

returns the vectors representation as a string (**repr** in python)

Parameters

in	*self*	the vector which data is to be presented in a readable way

Returns

the std::string containing the vector's representation

3.81.2.4 set()

```
static void VectorExposureVec3::set (
            std::vector< glm::vec3 > & self,
            uint32_t index,
            const bp::list & list )  [inline], [static]
```

sets the value at the given index of self to the given value

Parameters

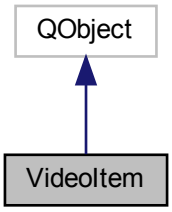
in,out	*self*	the vector which value at given index is to be changed
in	*index*	the index of the value to be changed
in	*list*	the list containing point coordinates to be applied to the point at the given index

The documentation for this class was generated from the following file:

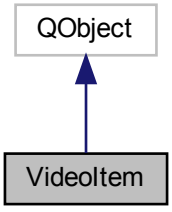
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/pysi/stl_container_exposure/VectorExposure.hpp

3.82 VideoItem Class Reference

Inheritance diagram for VideoItem:



Collaboration diagram for VideoItem:



Public Member Functions

- **VideoItem** (QObject ∗parent=nullptr)
- QAbstractVideoSurface ∗ **videoSurface** () const
- Q_SLOT void **setVideoSurface** (QAbstractVideoSurface ∗surface)
- Q_SLOT void **onVideoFrameReady** (QImage image)

Properties

- QAbstractVideoSurface **videoSurface**

The documentation for this class was generated from the following files:

- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/rendering/qml/items/VideoItem.hpp
- /home/juergen/1_dev/projects/Sketchable-Interaction/SI/src/sigrun/rendering/qml/items/VideoItem.cpp

Index