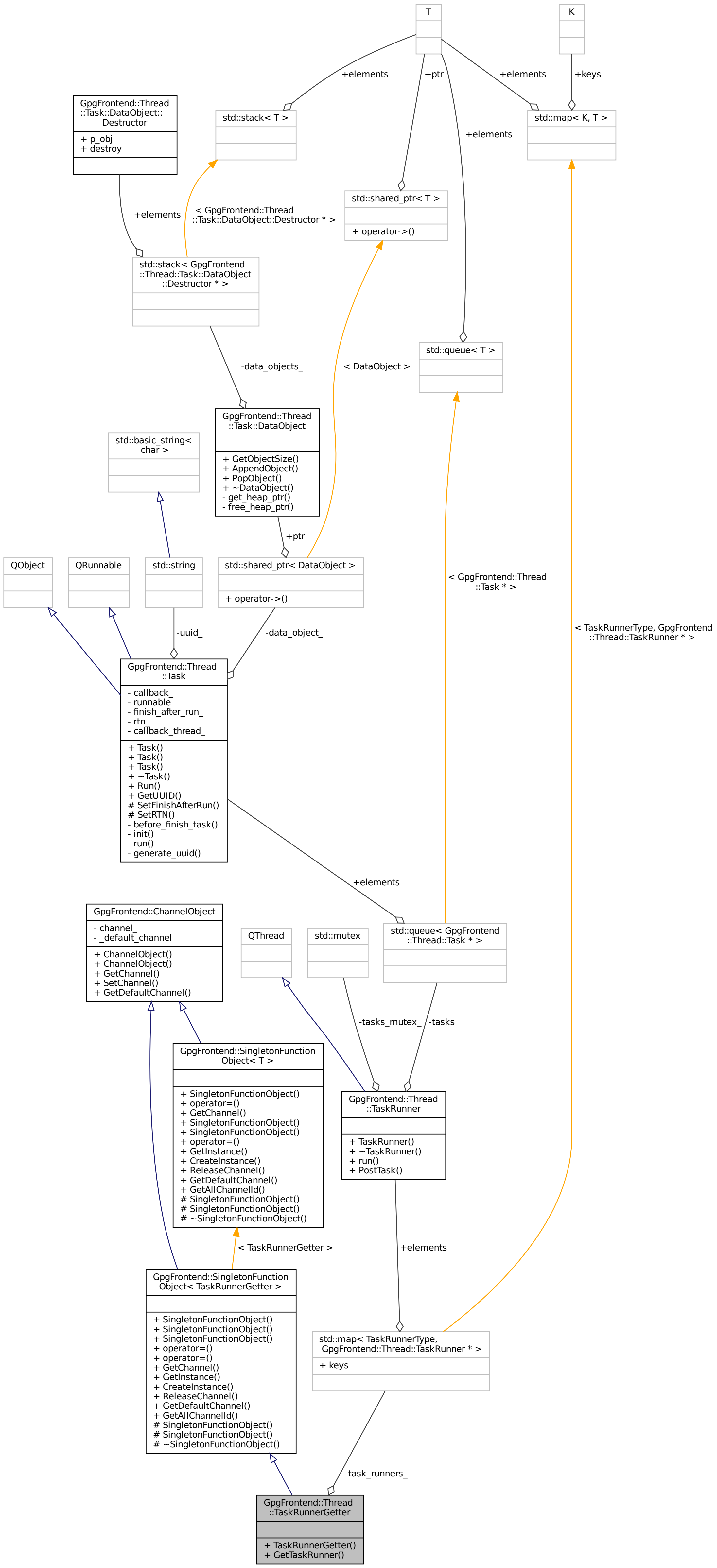
T
K
+elements
+ptr
+elements
+keys
GpgFrontend::Thread ::Task::DataObject:: Destructor
+ p_obj
+ destroy
std::stack< T >
std::map< K, T >
+elements
std::shared_ptr< T >
+ operator->()
+elements
< GpgFrontend::Thread ::Task::DataObject::Destructor * >
std::stack< GpgFrontend ::Thread::Task::DataObject ::Destructor * >
std::queue< T >
-data_objects_
< DataObject >
GpgFrontend::Thread ::Task::DataObject
+ GetObjectSize()
+ AppendObject()
+ PopObject()
+ ~DataObject()
- get_heap_ptr()
- free_heap_ptr()
std::basic_string< char >
+ptr
QObject
QRunnable
std::string
std::shared_ptr< DataObject >
+ operator->()
< GpgFrontend::Thread ::Task * >
-uuid_
-data_object_
< TaskRunnerType, GpgFrontend ::Thread::TaskRunner * >
GpgFrontend::Thread ::Task
- callback_
- runnable_
- finish_after_run_
- rtn_
- callback_thread_
+ Task()
+ Task()
+ Task()
+ ~Task()
+ Run()
+ GetUUID()
# SetFinishAfterRun()
# SetRTN()
- before_finish_task()
- init()
- run()
- generate_uuid()
+elements
GpgFrontend::ChannelObject
- channel_
- _default_channel
+ ChannelObject()
+ ChannelObject()
+ GetChannel()
+ SetChannel()
+ GetDefaultChannel()
QThread
std::mutex
std::queue< GpgFrontend ::Thread::Task * >
-tasks_mutex_
-tasks
GpgFrontend::SingletonFunction Object< T >
+ SingletonFunctionObject()
+ operator=()
+ GetChannel()
+ SingletonFunctionObject()
+ SingletonFunctionObject()
+ operator=()
+ GetInstance()
+ CreateInstance()
+ ReleaseChannel()
+ GetDefaultChannel()
+ GetAllChannelId()
# SingletonFunctionObject()
# SingletonFunctionObject()
# ~SingletonFunctionObject()
GpgFrontend::Thread ::TaskRunner
+ TaskRunner()
+ ~TaskRunner()
+ run()
+ PostTask()
< TaskRunnerGetter >
+elements
GpgFrontend::SingletonFunction Object< TaskRunnerGetter >
+ SingletonFunctionObject()
+ SingletonFunctionObject()
+ SingletonFunctionObject()
+ operator=()
+ operator=()
+ GetChannel()
+ GetInstance()
+ CreateInstance()
+ ReleaseChannel()
+ GetDefaultChannel()
+ GetAllChannelId()
# SingletonFunctionObject()
# SingletonFunctionObject()
# ~SingletonFunctionObject()
std::map< TaskRunnerType, GpgFrontend::Thread::TaskRunner * >
+ keys
-task_runners_
GpgFrontend::Thread ::TaskRunnerGetter
+ TaskRunnerGetter()
+ GetTaskRunner()