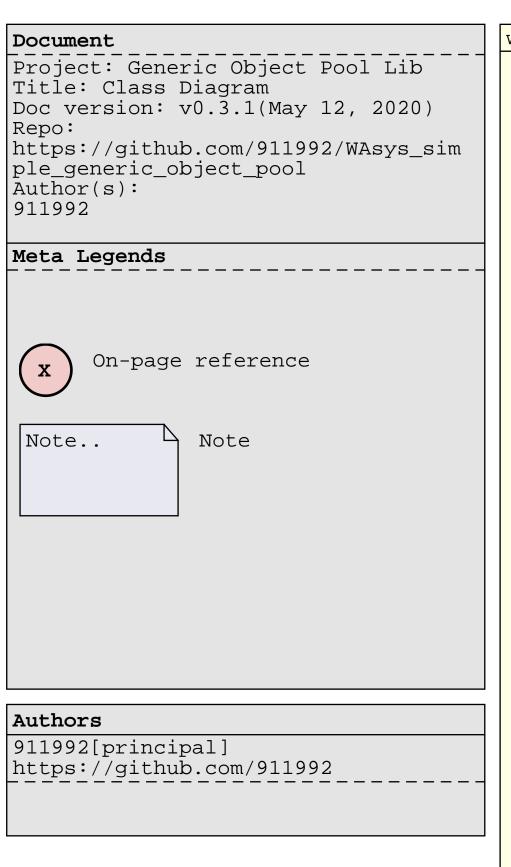
Version History v0.3.2 (May 15, 2020) Author: 911992 Fixed interface-interface inheritance notations(links) v0.3.1 (May 12, 2020) Author: 911992 Fixed ambiguous some fields access specifiers v0.3 (May 9, 2020) Author: 911992 Updated the note related to Generic_Object_Pool_Safe_Guard v0.2 (May 8, 2020) Author: 911992 Fixed some small notations Add missed singleton associate to Pool Context class Added Pool_Context::unregister_pool(:Obje ct Pool,:bool):void Mark register_pool(...):Object_Pool methods private, now use get_pool method instead Removed register_pool(...):Object_Pool method(s) Updated Object_Pool_Type_Wrapper constructor, to tell if pool object is needed to be registered or not Remove the confusing Object_Pool_Type_Wrapper constructor due to lack of doc in src code for now Added registered flag field in Generic_Object_Pool Fixed the misnamed Runnable as AutoClosable for Generic_Object_Pool Added is_registered(void):bool and get_policy(void):Generic_Object_Poo l_Policy methods to Object_Pool v0.1 (May 6, 2020) Author: 911992

Initial release



Document

Author(s):

Repo:

911992

Note..

Authors

```
wasys::lib::generic_object_pool
                                                                      «entity»
                                                             Generic_Object_Pool_Policy
+DEF MAX OBJ COUNT:int := 64
+DEF MIN OBJ COUNT:int := 8
+DEF OBJ CREATION POLICY: Full Pool Object Creation Policy := Create New No Pooling+
+DEF_INS:Generic_Object_Pool_Policy=new Generic_Object_Pool_Policy(DEF_MAX_OBJ_COUNT,DEF_MIN_OBJ_COUNT,DEF_OBJ_CREATION_POLICY)
+max_object_count:int {public-readonly} {package-settable}
+min_object_count:int
 -full_pool_instancing_policy:Full_Pool_Object_Creation_Policy
+Generic_Object_Pool_Policy(arg_max_obj_count:int,arg_min_obj_count:int,:Full_Pool_Object_Creation_Policy):Generic_Object_Pool_Policy
                                                                                A:Poolable_Object_
                                Object_Pool_Type_Wrapper
                                          {leaf}
-pool:Object_Pool {final}
+Object_Pool_Type_Wrapper(:Object_Factory,
                           :Generic_Object_Pool_Policy,
                                                                                                               Full_Pool_Object_Creation_Policy
                           arg_thread_safe:bool,
                                                                                                           Return_Null
                          arg_register_pool:bool):Object_Pool_Type_Wrapper
                                                                                                           Create_New_No_Pooling
                                                                                                           Create_New_Extend_Pool_Size
                                                                                                           Wait_Till_One_Free
      «creates»
                      | «uses»
                                                                                                                    «interface»
                                                                                                                    Object_Pool
                               «singleton(eager)»
                                                                                               +get_an_instance(void):Poolable_Object
                                 Pool_Context
                                                                                               +release_an_instance(:Poolable_Object):void
                                     {leaf}
                                                                                               +idle_objects_count(void):int
  -INSTANCE:Pool_Context = new Pool_Context()_ {final}
                                                                                               +available_objects_count(void):int
 ctx:Vector<Generic_Object_Pool> = new Vector(1,7) {final}
                                                                                               +working_object_count(void):int
 -Pool_Context(void):Pool_Context
                                                                                               +shutdown_pool(void):void
+get_instance(void):Pool_Context
                                                                                                +pool_is_working(void):bool
+get_pool_unregistered_synced(:Object_Factory,
:Generic_Object_Pool_Policy):Object_Pool {synchronized}
                                                                                               +is_registered(void):bool
                                                                                                +get_policy(void):Generic_Object_Pool_Policy
+get_pool_registered_synced(:Object_Factory,
                :Generic_Object_Pool_Policy):Object_Pool {synchronized}
+get_pool(:Object_Factory,:Generic_Object_Pool_Policy,
           arg thread safe:bool,
           arg_register:bool):Object_Pool {synchronized}
+unregister_pool(:Object_Pool,arg_shutdown_pool:bool):void {synchronized}
                                                                                                               «thread-safe»
                                                                                                       Generic_Object_Pool_Safe_Guard
                                                                                         -pool:Object_Pool {final}
                                                                ------
«creates»
                                                                                        ~Generic_Object_Pool_Safe(:Object_Pool,
                                                Lock on associated pool
                                                                                                                 ):Generic_Object_Pool_Safe
                                                may be applied by some
                                                methods
                «holds»
                                       В
                                                                                                                              E
                                                                  java::lang
                                                                                                                                    java::lang
                                                                  Runnable
                                                                                                                                    AutoClosable
                                                                             api
 i«creates»
                                                                                         «interface»
                                                                                                                               «abstract»
                                  «pool»
                           Generic Object Pool
                                                                                      Poolable Object
                                                                                                                        Poolable Object Adapter
                                                                                                              -factory:Object_Factory {final}
                                                                               +post_create(void):void
-policy:Generic_Object_Pool_Policy {final}
                                                                              +pre_destroy(void):void
                                                                                                                    +reset_state(void):void
-pool: Vector < Poolable Object >
                                                                              +reset_state(void):void
                                                                                                                    +close():void{final}
-working_ins_count:int := 0
                                                                              +set_pool(:Object_Pool):void
-pool working:bool := true {volatile}
-notify_thread_run:Runnable := new Runnable() {final}
-null_run:Runnable := new Runnable() {final}
-release_obj_run:Runnable {final}
                                                                                                |«creates»
-registered:bool
                                                                              «holds»
                                                                                                                        «interface»
~Generic_Object_Pool(:Object_Factory,
                                                                                                                       Object_Factory
                      :Generic_Object_Pool_Policy):Generic_Object_Pool
-init pool(void):void
                                                                                      «uses»
                                                                                                        +create_object(void):Poolable_Object
~get_factory(void):Object_Factory
~set_as_registered(:bool):void
~get_policy(void):Generic_Object_Pool_Policy
```