

MARTe2-core - User story #225

ClassRegistryDatabase unique identifier

07.09.2015 09:01 - André Neto

Status: Closed	Start date: 07.09.2015
Priority: Normal	Due date:
Assignee:	% Done: 0%
Category:	Estimated time: 0.00 hour
Target version: 0.5	Spent time: 0.00 hour
Git branch (link):	SVN commit (link/?p=rev):
Git merge to develop (link):	

Description

Refactor the ClassRegistryDatabase so that each registered class gets a unique identifier.

The implementation shall offer a search facility, by unique identifier, which allows to quickly retrieve a registered class (i.e. without having to walk all the list).

Tasks:

1. Develop a new list class StaticListHolder
2. Refactor the implementation of ClassRegistryDatabase (use StaticListHolder instead of LinkedListHolder as internal class database)

Source code files modified

- Source/Core/BareMetal/L1Portability/StaticList.cpp
- Source/Core/BareMetal/L1Portability/StaticList.h
- Source/Core/BareMetal/L1Portability/StaticListHolder.cpp
- Source/Core/BareMetal/L1Portability/StaticListHolder.h
- Source/Core/BareMetal/L1Portability/GlobalObjectI.cpp
- Source/Core/BareMetal/L1Portability/GlobalObjectI.h
- Source/Core/BareMetal/L1Portability/GlobalObjectsDatabase.cpp
- Source/Core/BareMetal/L1Portability/GlobalObjectsDatabase.h
- Source/Core/BareMetal/L1Portability/HeapManager.cpp
- Source/Core/BareMetal/L1Portability/HeapManager.h
- Source/Core/BareMetal/L2Objects/ClassRegistryDatabase.cpp
- Source/Core/BareMetal/L2Objects/ClassRegistryDatabase.h

- Test/Core/BareMetal/L1Portability/StaticListTest.cpp
- Test/Core/BareMetal/L1Portability/StaticListTest.h
- Test/Core/BareMetal/L1Portability/GlobalObjectsDatabaseTest.cpp
- Test/Core/BareMetal/L1Portability/GlobalObjectsDatabaseTest.h
- Test/Core/BareMetal/L2Objects/ClassRegistryDatabaseTest.cpp
- Test/Core/BareMetal/L2Objects/ClassRegistryDatabaseTest.h

- Test/GTest/StaticListGTest.cpp
- Test/GTest/GlobalObjectsDatabaseGTest.cpp

Architecture & design review

Date of the review: 08/09/2015

Person who did the review: Andre' Neto

Version of architecture & design document: N/A. As per the original MARTe implementation, this is a support function and does not require formal design in UML.

Result of review: N/A

List of non-conformities: N/A

Code and documentation review

Date of the review: 28/09/2015

Person who did the review: Ivan Herrero

Result of review: PASS

- In the StaticListHolder.h documentation, the formula MAXCAPACITY cannot be visualized with doxygen tool.
- GlobalObjectI is not an pure interface because the new operator is static and has an implementation.

List of non-conformities: N/A.

Unit test review

Date of the review: 28/09/2015

Person who did the review: Ivan Herrero

Result of coverage tests review: PASS

Result of functional tests review: PASS

Result of review: PASS

- TestAddOnFullList disabled due to it takes a quiet significant amount of time.
- The coverage is not 100% due to some conditions will never should happen.
- GlobalObjectsDatabaseTest lacks an explicit test for GlobalObjectsDatabase::Add, but it is covered indirectly by another tests.
- GlobalObjectI has not a 100% coverage because the compiler generates two versions of the destructor (one for delete and the other for scope).
- ClassRegistryDatabase has 11.9% lines not covered (this is was accepted on another user story review).

List of non-conformities: N/A

Associated revisions

Revision 3a754603 - 14.09.2015 11:29 - Ivan Herrero

#225 - Fix error on EclipseLint.sh script due to an incomplete merge on commit 41623e1

Revision 6f721f5f - 15.09.2015 18:10 - André Neto

#225 Finished implementation of ClassRegistryDatabase with StaticList.

As per story 225.2, the ClassRegistryDatabase logic was re-implemented using a StaticList as the backend engine. Linting was performed and the tests updated. This has to be merged with the work being developed in #225.1.

Revision ca5e6ba6 - 16.09.2015 17:52 - Ivan Herrero

#225 - Update flexlint configuration for StaticListHolder

Revision 505a3681 - 16.09.2015 17:59 - Ivan Herrero

#225 - Refactor, document, lint and fix bugs on StaticListHolder

Revision cfdceceb - 16.09.2015 18:01 - Ivan Herrero

#225 - First version of StaticList (extracted and refactored from StaticListHolder.h)

Revision 6dfa8b69 - 16.09.2015 18:02 - Ivan Herrero

#225 - Update of make configuration for StaticListHolder and StaticList

Revision d9896e3c - 16.09.2015 18:04 - Ivan Herrero

#225 - First version of StaticListTest

Revision 39baa8e9 - 16.09.2015 18:06 - Ivan Herrero

#225 - Update of make configuration for StaticListTest

Revision 9c9a2999 - 16.09.2015 18:07 - Ivan Herrero

#225 - First version of StaticListGTest

Revision 3eb76bd9 - 16.09.2015 18:07 - Ivan Herrero

#225 - Update of make configuration for StaticListGTest

Revision ea249788 - 21.09.2015 09:33 - Ivan Herrero

#225 - Evolution of StaticList

On StaticListHolder class:

- Correct spelling mistakes.
- Define some attributes as constants.
- Write a complete comment for the class.
- Improve and finish comments on methods and attributes.
- Implement destructor.
- Finish comments on method's implementations.
- Add systematic checks for preconditions.

On StaticListTest class:

- Correct spelling mistakes.
- Improve tests for GetSize and DefaultConstructor

Revision 2bc6bc45 - 22.09.2015 09:56 - Ivan Herrero

#225 - Add tests and format file StaticListGTest.cpp

Revision 7f580546 - 22.09.2015 10:17 - Ivan Herrero

#225 - Finish StaticList classes

Add documentation and methods on StaticList and StaticListTest class.
Remove explicit instantiations for StaticList and StaticListTest class.
Add Remove method on StaticListHolder class.
Apply lint corrections/exceptions on StaticListHolder class.

Revision 1b490d07 - 22.09.2015 10:49 - Ivan Herrero

Merge remote-tracking branch 'origin/#225.1_StaticListHolder' into #225.2_StaticListHolder

Conflicts:

Source/Core/L0TypeDev/StaticListHolder.cpp
Source/Core/L0TypeDev/StaticListHolder.h

Revision fd0a4db0 - 22.09.2015 11:07 - Ivan Herrero

Merge branch '#225.2_StaticListHolder' of <https://vcis-gitlab.f4e.europa.eu/aneto/MARTE2.git> into #225.2_StaticListHolder

Conflicts:

Source/Core/L1Objects/ClassRegistryDatabase.cpp
Test/GTest/ClassRegistryDatabaseGTest.cpp

Revision 53c70587 - 22.09.2015 11:30 - Ivan Herrero

Merge with #225.1 and old #225.2

Revision 6b3cdb74 - 23.09.2015 17:40 - André Neto

#225 Changed to the folder structure of #226.

Revision e17de0c4 - 23.09.2015 18:50 - André Neto

Merge branch '#197_Any_Type_Conversion' into #225.2_StaticListHolder

Conflicts:

EclipseLint.sh

MakeDefaults/Lint/marte_flint_base_configuration.Int

MakeDefaults/Lint/marte_flint_exceptions.Int

Source/Core/BareMetal/L0Types/BasicType.h

Source/Core/BareMetal/L0Types/BitBoolean.h

Source/Core/BareMetal/L0Types/BitRange.h

Source/Core/BareMetal/L0Types/CompilerTypes.h

Source/Core/BareMetal/L0Types/ErrorType.h

Source/Core/BareMetal/L0Types/FlagsType.h

Source/Core/BareMetal/L0Types/FractionalInteger.h

Source/Core/BareMetal/L0Types/Introspection.h

Source/Core/BareMetal/L0Types/Iterator.h

Source/Core/BareMetal/L0Types/IteratorT.h

Source/Core/BareMetal/L0Types/LinkedListHolder.cpp

Source/Core/BareMetal/L0Types/LinkedListHolder.h

Source/Core/BareMetal/L0Types/LinkedListable.cpp

Source/Core/BareMetal/L0Types/LinkedListable.h

Source/Core/BareMetal/L0Types/SearchFilter.h

Source/Core/BareMetal/L0Types/SearchFilterT.h

Source/Core/BareMetal/L0Types/SortFilter.h

Source/Core/BareMetal/L0Types/SortFilterT.h

Source/Core/BareMetal/L0Types/StructuredData.h

Source/Core/BareMetal/L0Types/TypeCharacteristics.h

Source/Core/BareMetal/L0Types/TypeDescriptor.h

Source/Core/BareMetal/L1Portability/Architecture/x86_gcc/AtomicA.h

Source/Core/BareMetal/L1Portability/Architecture/x86_gcc/EndianityA.h

Source/Core/BareMetal/L1Portability/Architecture/x86_gcc/FastMathA.h

Source/Core/BareMetal/L1Portability/Architecture/x86_gcc/HighResolutionTimerA.h

Source/Core/BareMetal/L1Portability/Architecture/x86_gcc/ProcessorA.h

Source/Core/BareMetal/L1Portability/Atomic.h

Source/Core/BareMetal/L1Portability/BasicConsole.h

Source/Core/BareMetal/L1Portability/Endianity.h

Source/Core/BareMetal/L1Portability/Environment/Generic/MemoryOperationsHelper_CLIB_Generic.cpp

Source/Core/BareMetal/L1Portability/Environment/Generic/StandardHeap_Generic.cpp

Source/Core/BareMetal/L1Portability/Environment/Generic/StandardHeap_Generic.h

Source/Core/BareMetal/L1Portability/Environment/Generic/StringHelperExtras_Generic.cpp

Source/Core/BareMetal/L1Portability/Environment/Generic/StringHelper_CLIB_Generic.cpp

Source/Core/BareMetal/L1Portability/Environment/Generic/StringHelper_Portable_Generic.cpp

Source/Core/BareMetal/L1Portability/Environment/Linux/BasicConsole.cpp

Source/Core/BareMetal/L1Portability/Environment/Linux/GeneralDefinitions.h

Source/Core/BareMetal/L1Portability/Environment/Linux/HighResolutionTimer.h

Source/Core/BareMetal/L1Portability/Environment/Linux/HighResolutionTimerCalibrator.cpp

Source/Core/BareMetal/L1Portability/Environment/Linux/HighResolutionTimerCalibrator.h

Source/Core/BareMetal/L1Portability/Environment/Linux/LoadableLibrary.cpp

Source/Core/BareMetal/L1Portability/Environment/Linux/StandardHeap.h

Source/Core/BareMetal/L1Portability/Environment/Linux/lint-linux.h

Source/Core/BareMetal/L1Portability/ErrorInformation.h

Source/Core/BareMetal/L1Portability/ErrorMessageManagement.cpp

Source/Core/BareMetal/L1Portability/ErrorMessageManagement.h

Source/Core/BareMetal/L1Portability/FastMath.h

Source/Core/BareMetal/L1Portability/FastPollingMutexSem.h

Source/Core/BareMetal/L1Portability/FormatDescriptor.cpp

Source/Core/BareMetal/L1Portability/FormatDescriptor.h

Source/Core/BareMetal/L1Portability/HeapI.h

Source/Core/BareMetal/L1Portability/HeapManager.cpp

Source/Core/BareMetal/L1Portability/HeapManager.h

Source/Core/BareMetal/L1Portability/LoadableLibrary.h

Source/Core/BareMetal/L1Portability/MemoryCheck.h

Source/Core/BareMetal/L1Portability/MemoryOperationsHelper.h

Source/Core/BareMetal/L1Portability/ProcessorType.cpp

Source/Core/BareMetal/L1Portability/ProcessorType.h

Source/Core/BareMetal/L1Portability/Sleep.h

Source/Core/BareMetal/L1Portability/StaticListHolder.cpp

Source/Core/BareMetal/L1Portability/StaticListHolder.h

Source/Core/BareMetal/L1Portability/StringHelper.cpp

Source/Core/BareMetal/L1Portability/StringHelper.h

Source/Core/BareMetal/L1Portability/TimeoutType.h

Source/Core/BareMetal/L2Objects/ClassProperties.cpp

Source/Core/BareMetal/L2Objects/ClassProperties.h

Source/Core/BareMetal/L2Objects/ClassRegistryDatabase.cpp

Source/Core/BareMetal/L2Objects/ClassRegistryDatabase.h

Source/Core/BareMetal/L2Objects/ClassRegistryItem.cpp

Source/Core/BareMetal/L2Objects/ClassRegistryItem.h

Source/Core/BareMetal/L2Objects/Object.cpp

Source/Core/BareMetal/L2Objects/Object.h
Source/Core/BareMetal/L2Objects/Reference.cpp
Source/Core/BareMetal/L2Objects/Reference.h
Source/Core/BareMetal/L2Objects/ReferenceContainer.cpp
Source/Core/BareMetal/L2Objects/ReferenceContainer.h
Source/Core/BareMetal/L2Objects/ReferenceContainerFilter.cpp
Source/Core/BareMetal/L2Objects/ReferenceContainerFilter.h
Source/Core/BareMetal/L2Objects/ReferenceContainerFilterObjectName.cpp
Source/Core/BareMetal/L2Objects/ReferenceContainerFilterObjectName.h
Source/Core/BareMetal/L2Objects/ReferenceContainerFilterReferences.cpp
Source/Core/BareMetal/L2Objects/ReferenceContainerFilterReferences.h
Source/Core/BareMetal/L2Objects/ReferenceContainerNode.cpp
Source/Core/BareMetal/L2Objects/ReferenceContainerNode.h
Source/Core/BareMetal/L2Objects/ReferenceT.cpp
Source/Core/BareMetal/L2Objects/ReferenceT.h
Source/Core/BareMetal/L2Objects/AnyType.h
Source/Core/Scheduler/L1Portability/Environment/Generic/ThreadsDatabase_Generic.cpp
Source/Core/Scheduler/L1Portability/Environment/Generic/ThreadsDatabase_Generic.h
Source/Core/Scheduler/L1Portability/Environment/Linux/EventSem.cpp
Source/Core/Scheduler/L1Portability/Environment/Linux/MutexSem.cpp
Source/Core/Scheduler/L1Portability/Environment/Linux/Processor.cpp
Source/Core/Scheduler/L1Portability/Environment/Linux/Threads.cpp
Source/Core/Scheduler/L1Portability/Environment/Windows/Sleep.cpp
Source/Core/Scheduler/L1Portability/EventSem.h
Source/Core/Scheduler/L1Portability/ExceptionHandler.h
Source/Core/Scheduler/L1Portability/MutexSem.h
Source/Core/Scheduler/L1Portability/Processor.h
Source/Core/Scheduler/L1Portability/ThreadInformation.cpp
Source/Core/Scheduler/L1Portability/ThreadInformation.h
Source/Core/Scheduler/L1Portability/Threads.h
Test/Core/BareMetal/L0Types/BasicTypeTest.cpp
Test/Core/BareMetal/L0Types/BitBooleanTest.h
Test/Core/BareMetal/L0Types/BitRangeTest.h
Test/Core/BareMetal/L0Types/FractionalIntegerTest.h
Test/Core/BareMetal/L0Types/LinkedListHolderTest.cpp
Test/Core/BareMetal/L0Types/LinkedListHolderTest.h
Test/Core/BareMetal/L0Types/LinkedListableTest.cpp
Test/Core/BareMetal/L0Types/LinkedListableTest.h
Test/Core/BareMetal/L0Types/ListTestHelper.h
Test/Core/BareMetal/L0Types/TypeCharacteristicsTest.cpp
Test/Core/BareMetal/L0Types/TypeDescriptorTest.cpp
Test/Core/BareMetal/L0Types/TypeDescriptorTest.h
Test/Core/BareMetal/L1Portability/AtomicTest.h
Test/Core/BareMetal/L1Portability/AuxHeap.cpp
Test/Core/BareMetal/L1Portability/AuxHeap.h
Test/Core/BareMetal/L1Portability/BasicConsoleTest.cpp
Test/Core/BareMetal/L1Portability/BasicConsoleTest.h
Test/Core/BareMetal/L1Portability/EndiannessTest.h
Test/Core/BareMetal/L1Portability/ErrorMessageTest.cpp
Test/Core/BareMetal/L1Portability/ErrorMessageTest.h
Test/Core/BareMetal/L1Portability/FastMathTest.h
Test/Core/BareMetal/L1Portability/FastPollingMutexSemTest.h
Test/Core/BareMetal/L1Portability/FormatDescriptorTest.cpp
Test/Core/BareMetal/L1Portability/FormatDescriptorTest.h
Test/Core/BareMetal/L1Portability/HeapManagerTest.cpp
Test/Core/BareMetal/L1Portability/HeapManagerTest.h
Test/Core/BareMetal/L1Portability/HighResolutionTimerTest.cpp
Test/Core/BareMetal/L1Portability/HighResolutionTimerTest.h
Test/Core/BareMetal/L1Portability/LoadableLibraryTest.cpp
Test/Core/BareMetal/L1Portability/LoadableLibraryTest.h
Test/Core/BareMetal/L1Portability/Makefile.inc
Test/Core/BareMetal/L1Portability/MemoryCheckTest.cpp
Test/Core/BareMetal/L1Portability/MemoryIntegrationTest.h
Test/Core/BareMetal/L1Portability/MemoryOperationsHelperTest.cpp
Test/Core/BareMetal/L1Portability/MemoryTest.h
Test/Core/BareMetal/L1Portability/ProcessorTypeTest.h
Test/Core/BareMetal/L1Portability/SleepTest.cpp
Test/Core/BareMetal/L1Portability/SleepTest.h
Test/Core/BareMetal/L1Portability/StandardHeap_GenericTest.cpp
Test/Core/BareMetal/L1Portability/StandardHeap_GenericTest.h
Test/Core/BareMetal/L1Portability/StringHelperTest.h
Test/Core/BareMetal/L1Portability/StringPortableTest.h
Test/Core/BareMetal/L1Portability/StringTestHelper.h
Test/Core/BareMetal/L1Portability/TimeoutTypeTest.cpp
Test/Core/BareMetal/L1Portability/TimeoutTypeTest.h

Test/Core/BareMetal/L2Objects/ClassPropertiesTest.cpp
Test/Core/BareMetal/L2Objects/ClassPropertiesTest.h
Test/Core/BareMetal/L2Objects/ClassRegistryDatabaseTest.cpp
Test/Core/BareMetal/L2Objects/ClassRegistryDatabaseTest.h
Test/Core/BareMetal/L2Objects/ClassRegistryItemTest.cpp
Test/Core/BareMetal/L2Objects/ClassRegistryItemTest.h
Test/Core/BareMetal/L2Objects/ObjectTest.h
Test/Core/BareMetal/L2Objects/ObjectTestHelper.h
Test/Core/BareMetal/L2Objects/ReferenceContainerFilterObjectNameTest.h
Test/Core/BareMetal/L2Objects/ReferenceContainerFilterReferencesTest.cpp
Test/Core/BareMetal/L2Objects/ReferenceContainerFilterReferencesTest.h
Test/Core/BareMetal/L2Objects/ReferenceContainerNodeTest.cpp
Test/Core/BareMetal/L2Objects/ReferenceContainerTest.h
Test/Core/BareMetal/L2Objects/ReferenceTTest.cpp
Test/Core/BareMetal/L2Objects/ReferenceTTest.h
Test/Core/BareMetal/L2Objects/ReferenceTest.cpp
Test/Core/BareMetal/L2Objects/ReferenceTest.h
Test/Core/BareMetal/L2Objects/TestObjectHelper1.h
Test/Core/BareMetal/L2Objects/TestObjectHelper2.h
Test/Core/BareMetal/L3Streams/AnyTypeTest.cpp
Test/Core/BareMetal/L3Streams/AnyTypeTest.h
Test/Core/Scheduler/L1Portability/EventSemTest.cpp
Test/Core/Scheduler/L1Portability/EventSemTest.h
Test/Core/Scheduler/L1Portability/MutexSemTest.cpp
Test/Core/Scheduler/L1Portability/MutexSemTest.h
Test/Core/Scheduler/L1Portability/ProcessorTest.cpp
Test/Core/Scheduler/L1Portability/ProcessorTest.h
Test/Core/Scheduler/L1Portability/ThreadsTest.h

Revision 75f6a706 - 27.09.2015 10:04 - André Neto

#225 Implementation of GlobalObjectsDatabase

The implementation of the ClassRegistryDatabase with a StaticList backend has generated a racing condition between the HeapDatabase and ClassRegistryDatabase singletons. This was making the test program crash at the end of its execution. As a consequence it was possible to run the tests for this story.

To address the issue above a global object manager, named GlobalObjectsDatabase, was implemented. This will be the only true global object in the framework. All the other singleton objects will be dynamically allocated in the heap and registered in this GlobalObjectDatabase.

Revision 215e272a - 28.09.2015 16:53 - Ivan Herrero

#225 - Review and fix source and test files

- Apply formatting rules and fix documentation issues.
- Improve documentation for MARTe::GlobalObjectsDatabase::Add
- Remove "using namespace MARTe;" from ClassRegistryDatabaseTest.h because is a header file and this pollutes the namespace of the users of the header.

History

#1 - 08.09.2015 12:52 - André Neto

- *Description updated*
- *Status changed from New to Code: Impl*

#2 - 08.09.2015 12:53 - André Neto

- *Target version set to 0.5*

#3 - 08.09.2015 12:56 - André Neto

- *Description updated*

#4 - 09.09.2015 09:30 - Ivan Herrero

- *Description updated*

#5 - 15.09.2015 16:16 - André Neto

- Assignee set to Ivan Herrero

#6 - 22.09.2015 11:35 - Ivan Herrero

- Status changed from Code: Impl to Unit: Rev

#7 - 22.09.2015 11:35 - Ivan Herrero

- Status changed from Unit: Rev to Code: Rev

#8 - 22.09.2015 11:35 - Ivan Herrero

- Assignee deleted (Ivan Herrero)

#9 - 22.09.2015 16:01 - Llorenc Capella

- Assignee set to Llorenc Capella

#10 - 23.09.2015 10:30 - Llorenc Capella

- Description updated

- Assignee deleted (Llorenc Capella)

#11 - 23.09.2015 11:05 - Ivan Herrero

- Status changed from Code: Rev to Unit: Rev

#12 - 23.09.2015 12:38 - Llorenc Capella

- Assignee set to Llorenc Capella

#13 - 23.09.2015 16:30 - Llorenc Capella

- Description updated

- Assignee deleted (Llorenc Capella)

#14 - 27.09.2015 09:59 - André Neto

- Status changed from Unit: Rev to Code: Rev

#15 - 27.09.2015 10:00 - André Neto

The implementation of the ClassRegistryDatabase with a StaticList backend has generated a racing condition between the HeapDatabase and ClassRegistryDatabase singletons. This was making the test program crash at the end of its execution. As a consequence it was not possible to run the tests for this story.

To address the issue above a global object manager, named GlobalObjectsDatabase, was implemented. This will be the only true global object in the framework. All the other singleton objects will be dynamically allocated in the heap and registered in this GlobalObjectDatabase.

The story was moved back to code review so that the implementation of the GlobalObjectsDatabase can be reviewed.

#16 - 27.09.2015 10:05 - André Neto

- Description updated

#17 - 28.09.2015 14:39 - Ivan Herrero

- Description updated

#18 - 28.09.2015 17:01 - Llorenc Capella

- Status changed from Code: Rev to Unit: Rev

#19 - 28.09.2015 17:05 - Ivan Herrero

- Description updated

#20 - 28.09.2015 18:12 - Ivan Herrero

- Status changed from Unit: Rev to Closed

Merged on develop.

#21 - 29.09.2015 09:47 - Ivan Herrero

- Description updated