UCARE Research

Ray Andrew

October 8, 2019

Abstract

The abstract text goes here.

Contents

0.1 Todos

0.1.1 Subsect

Write your subsection text here.

0.2 Summarization

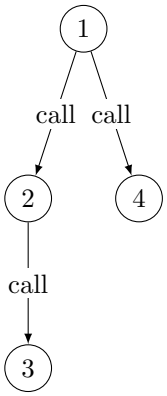
0.2.1 Subsect

Write your subsection text here.

0.3 Notes

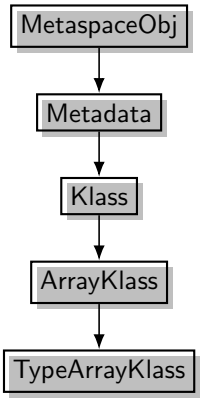
0.3.1 Monday, 07 October 2019

0.3.1.1 JVM Array Allocation



1. `void OptoRuntime::new_array_C(Klass* array_type, int len, JavaThread *thread)`
2. `typeArrayOop oopFactory::new_typeArray(BasicType type, int length, TRAPS)`
3. `TypeArrayKlass* TypeArrayKlass::allocate(ClassLoaderData* loader_data, BasicType type` -¿ new instance of TypeArrayKlass
4. `TypeArrayKlass* TypeArrayKlass::allocate(ClassLoaderData* loader_data, BasicType type` -¿ new instance of TypeArrayKlass
5. `objArrayOop oopFactory::new_objArray(Klass* klass, int length, TRAPS)`

0.3.1.2 Type Array Inheritance



0.3.2 Friday, 04 October 2019

0.3.2.1 Allocation Classes

1. Allocation, consists of:
1.1. AllocatedObj (Abstract Class)
1.1.1. CHeapObj (Abstract Class)
1.1.1.1. CollectedHeap
1.1.1.1.1. GenCollectedHeap
1.1.1.1.1.1. SerialHeap
1.1.1.1.1.2. CMSHeap
1.1.1.1.2. G1CollectedHeap
1.1.1.1.3. ParallelScavengeHeap
1.1.1.1.4. ZCollectedHeap
1.1.2. StackObj (Abstract Class)
1.1.2.1. MemAllocator
1.1.2.1.1. ObjAllocator
1.1.2.1.2. ObjArrayAllocator
1.1.2.1.3. ClassAllocator
1.1.3. ResourceObj (Abstract Class)
1.2. MetaspaceObj (Abstract Class)
1.3. AllStatic (Abstract Class)
1.3.1. ArrayAllocator
1.3.2. MmapArrayAllocator
1.4. MallocArrayAllocator

0.3.2.2 Notes

- All of the Klass is being instantiated in Metaspace
- EpsilonGC is not creating its own heap

0.4 Resources

0.4.1 Kernel Samepage Merging

1. Documentation
   (a) Madvise
   (b) KSM