JNode Isolates & GC

Long term & short term
Goals & ideas

Contents

- Goals
- Global architecture
- Design decisions
- Short term
  - Goals
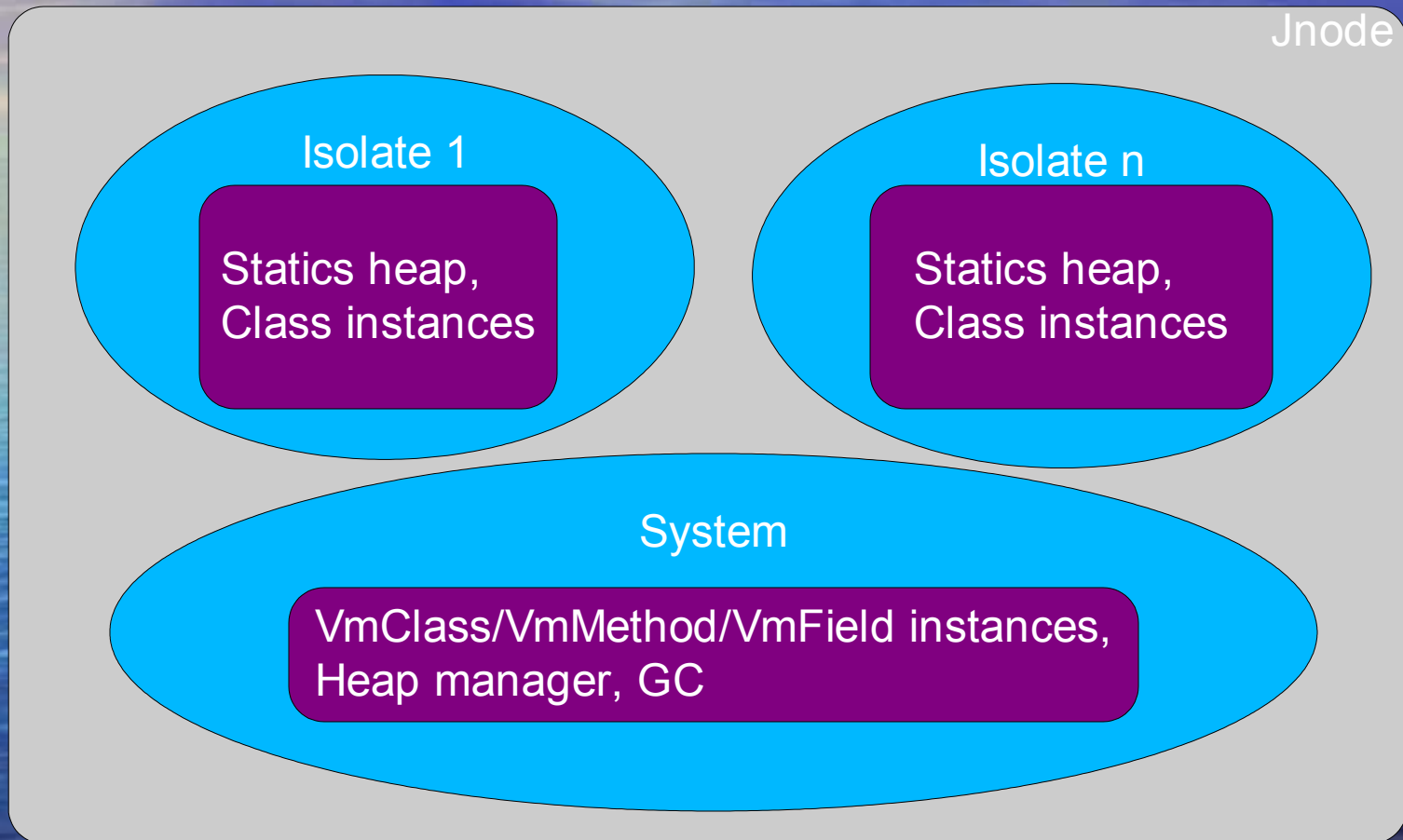 - GC
  - Write Barrier

Goals

- Support JSR 121
- "Implement" processes using Isolates
- Hardware architecture independent
- Isolates should be light weight

Global architecture

Jnode

Isolate 1

Statics heap,
Class instances

Isolate n

Statics heap,
Class instances

System

VmClass/VmMethod/VmField instances,
Heap manager, GC

Design decisions

- Support multiple Isolate's in a single VM
- Stick to single flat memory space

Questions

- Single GC / GC per Isolate
- What to do with driver framework:
  - In single Isolate & message passing or
  - Shared across Isolates
- How to differentiate between
  - Shared statics (internals of the VM)
  - Non-shared statics

Short term: Goals

- Implement concurrent mark & sweep gc
- Prepare for Isolates

Short term: GC

- Mark phase concurrent with mutator
  - Protect object tree with Write Barrier
- Separate GC thread:
  - Mark phase
  - Sweep phase
  - Optimize phase (join adjacent free blocks)

Short term: Write Barrier

- Use tri-color abstraction
- Newly allocated objects are "black"
- First implement in java (for testability)
  - Maybe later in native code, but what about the interpreter