



Recent & Incoming Cog VM improvements

Clément Béra



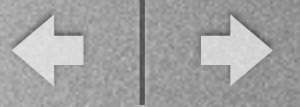
Spur Memory manager

- 1) Faster become
- 2) Ephemerons
- 3) Pinned objects
- 4) Segmented memory
- 5) Object format 64 bits compatible
- 6) Incremental and efficient GC
- 7) Performance improvement



Faster become

- Old implementation: need a full heap scan
- New implementation: need **only** a stack zone scan
- Forwarding pointers



Ephemeron

- Improved weak structures



Pinned objects

- Mark an object so it won't move in memory
- Very useful, for example, for FFI calls



Segmented memory

- Heap used to be contiguous with its issues: memory limits, important allocation
- Heap is now chopped into pieces:
 - dynamically grows and shrinks
 - better interaction with the OS



Object format 64 bits

- **First step towards a 64 bits Pharo**



Incremental / Efficient GC

- Incremental GC: split the GC work in order to limit the user pauses (partially done)
- Efficient GC: Multiple generation scavenging



Performance

- - 50% (x2 speed up)
 - faster GC
 - faster instantiation
 - faster at:put:



Spur Memory manager

- Alpha version
- Integration in ?



64 bits

- 64 bits spur stack VM running on Linux
- Fast floating pointers
- JIT, FFI, ... to port



ARM improvements

- FFI on ARM
- VM on ARMv6, ARMv7: Raspberry pie, Android, ...
- (Jean-Baptiste)



New Bytecode set

- Bytecode set supporting Smalltalk /
bytecode generation
- Fewer encoding limits
- Integration in ?



Questions