

Powerlang: a Vehicle for Lively Implementing Programming Languages



Javier Pimás
Guido Chari

What does it take to create a PL?

- Compiler
 - Parser
 - Scanner
 - Execution Semantics
 - Assembler
- Virtual Machine
 - Primitives
 - Garbage Collection

What does it take to create a PL?

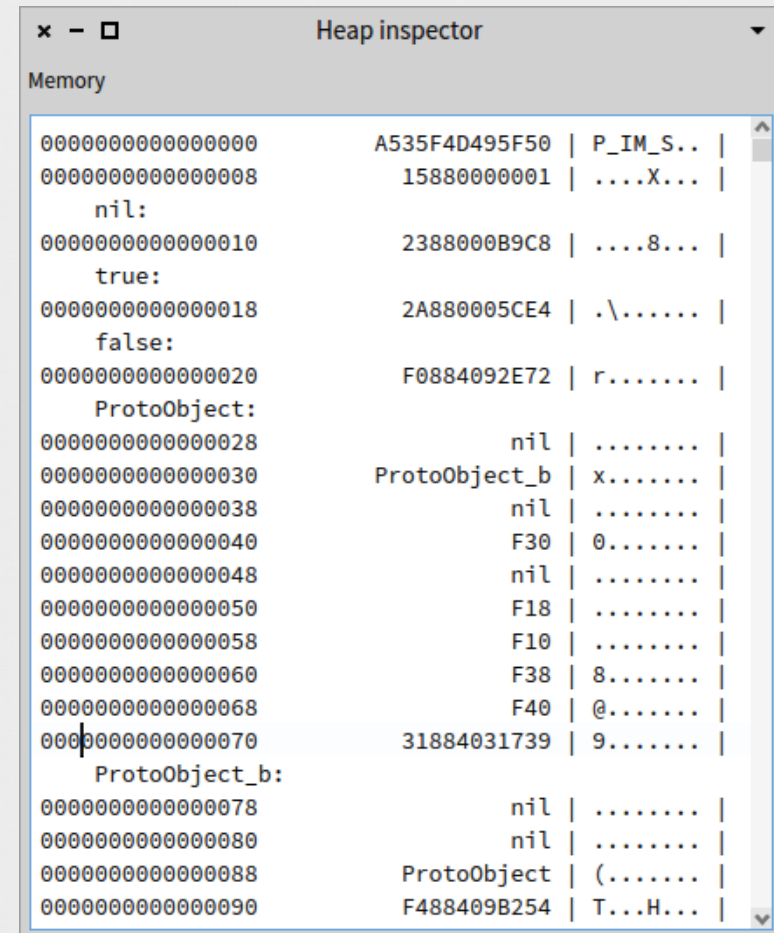
- Bootstrapping
 - Executable image
- Debugging
 - Remote execution
 - Simulation

Current approaches

- Other existing VM
- Metacompilation frameworks
- Micro VMs
- Write your own
- Powerlang

Bootstrapping

- Minimal initial effort
- Bit-by-bit reproducibility
- Minimal deploy size



Simulation and Debugging

- Incomplete systems should be debuggable
- Both high- and low-level debugging

The screenshot displays the Powerleng Debugger interface. The top toolbar includes buttons for 'Proceed', 'Restart', 'Step Into', 'Step Through', 'Step Over', and 'Step Out'. The main window is divided into three panes:

- Left Pane (Ruby Code):** Shows the execution of the `basicAdd` method on an `anObject`. The code includes `end = contents size`, `end := end + 1.`, and `contents at: end put: anObject.`. The current execution point is at `^anObject`.
- Middle Pane (Assembly):** Displays assembly instructions. The current instruction is `19: call QWORD PTR [rdx]`, which is highlighted in blue. Other instructions include `0: push rbp`, `1: mov rbp, rsp`, `4: push rax`, `5: mov rsi, rax`, `8: push QWORD PTR [rsi+0x8]`, `b: mov rax, QWORD PTR [rsi+0x10]`, and `1b: mov rdx, rax`.
- Right Pane (Backtrace):** Shows the call stack. The current frame is `OrderedCollection>>basicAdd: 1`. Other frames include `OrderedCollection>>add: 1`, `Object>>foo []>>`, `Array(SequenceableCollection)>>do:`, `Object>>foo`, and `Object>>doit`.

Compiling and Optimizing

- Provide compilation tooling
- How to model of language behavior?

m

^self foo

#(1 #(9 true #(5 1 #(3 6))))

Current and Future work

github.com/melkyades/powerlang

- MIT license
- Initial proof-of-concept Smalltalk code
- Two VM implementations
 - DMR fully dynamic VM
 - eclipse OMR

Questions?