

MAXINE VM INTRODUCTION

- Meta-circular VM (Java VM written in Java)
- Multiple JIT compilers (T1X, C1X, Graal*)
 - JVMCI Compatibility*
- Multiple GC algorithms
 - MMTk Integration*
- Multiple ISAs
 - X86_64, ARMv7, Aarch64, RISC-V*
- Cross-ISA testing framework [1]
 - Allow porting of compilers on new ISAs through cross-compilation and simulation
- Integration with ZSim in MaxSim [2]
- Part of the Beehive ecosystem [3]

[1] C. Kotselidis, A. Nisbet, F. S. Zakkak, N. Fouttris. Cross-ISA debugging in meta-circular VMs. In VMIL 2017.

[2] A. Rodchenko, C. Kotselidis, A. Nisbet, A. Pop, M. Lujan. MaxSim: A simulation platform for managed applications. In ISPASS 2017.

[3] C. Kotselidis, J. Clarkson, A. Rodchenko, A. Nisbet, J. Mawer, M. Luján. Heterogeneous Managed Runtime Systems: A Computer Vision Case Study. In VEE 2017.

* Work in progress

MAXINE VM RISC-V STATUS

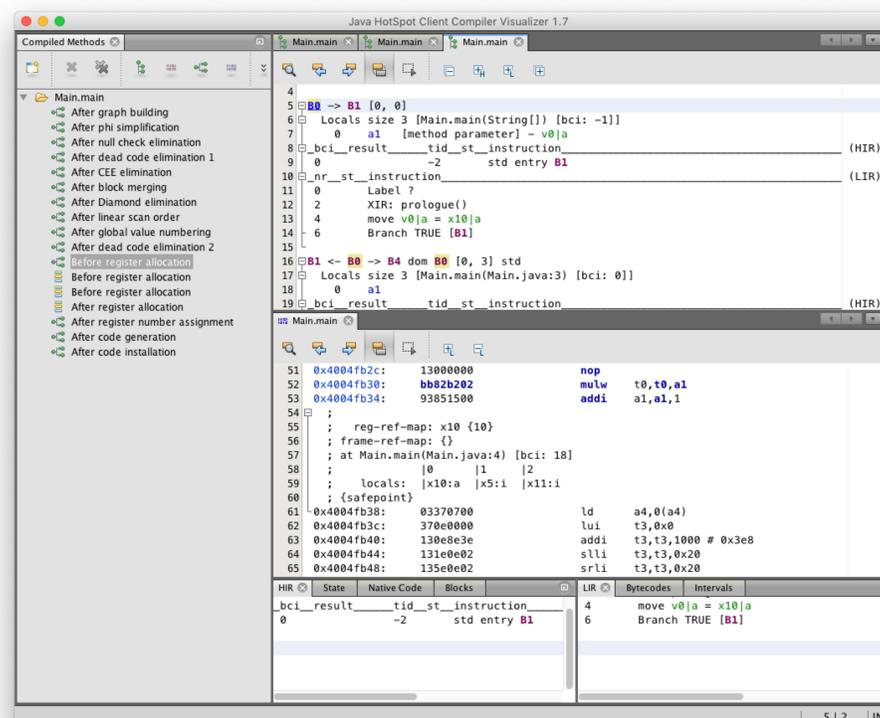
- ✓ Cross-ISA Testing Framework
- ✓ RISC-V Assembler
- ✓ T1X Baseline Compiler
- ✓ C1X Optimizing Compiler
- Boot Image Generation

GENERATING RISC-V CODE

```
$ mx --J @"-Dmax.platform=linux-riscv64" o1c -c=C1X -cp ./ -C1X:+PrintLIRWithAssembly -C1X:+PrintCFGToFile ^Main:main^
...
B2 (bv)[11, 18] . B4 dom B4 pred: B4 24 Label ?
0xb8: 13000000 nop
0xbc: 13000000 nop
28 Mul x5:i = (x5:i, x11:i)
0xc0: bb82b202 mulw t0,t0,a1
32 Add x11:i = (x11:i, const[1|0x1]:i)
0xc4: 93851500 addi a1,a1,1
34 XIR: safepoint() temp=(r14:j) [bci:18, refmap(x10:a)]
0xc8: 03370700 ld a4,0(a4)
...
$ ls o1c/* compilations-*
compilations-1557493502272.cfg Main.main-C1X-riscv64.S Main.main-C1X-riscv64.bin
```

```
public class Main {
    Run | Debug
    public static void main(String[] args) {
        int result = 1;
        for (int i = 1; i < 1000; i++) {
            result *= i;
        }
        System.out.println(result);
    }
}
```

VISUALIZING COMPILATION



ARTIFACTS

- *.cfg can be visualized with c1visualizer
- *.S contain the disassembled generated code
- *.bin contain the generated raw machine code for the corresponding method

CONTACT

All software is open-source:
<https://github.com/beehive-lab>

We welcome external contributions!
Please contact:
christos.kotselidis@manchester.ac.uk

