

# **Pydgin for RISC-V: A Fast and Productive Instruction-Set Simulator**

Berkin Ibeyi

In collaboration with Derek Lockhart (Google), and Christopher Batten

3rd RISC-V Workshop, Jan 2016



Cornell University  
Computer Systems Laboratory

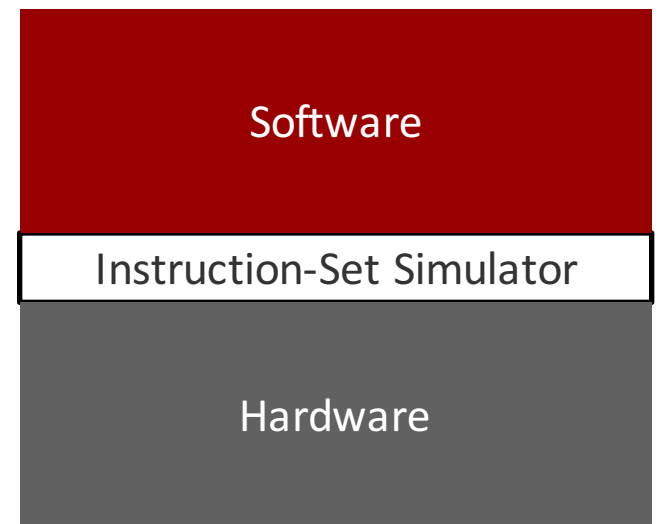
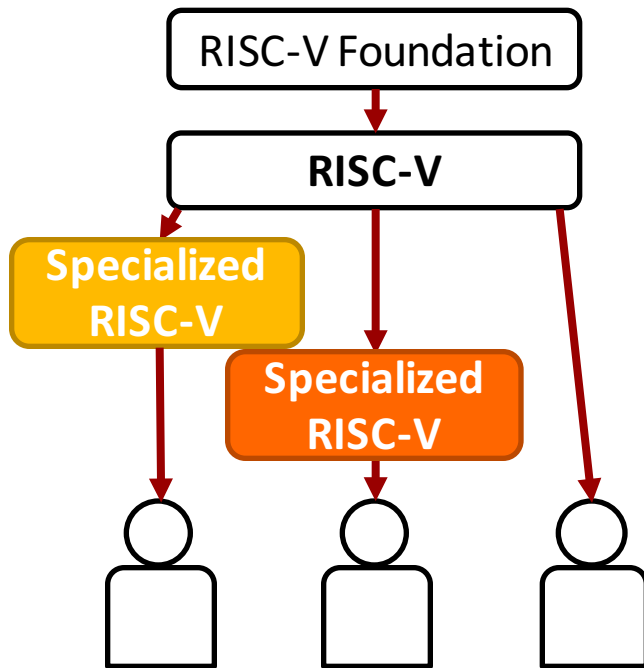
# Motivation

## Productivity

- Develop
- Extend
- Instrument

## Performance

Interpretive: 1-10 MIPS (1-10 days)  
Typical DBT: 100s MIPS (1-3 hours)  
QEMU DBT: 1000 MIPS (0.5 hours)

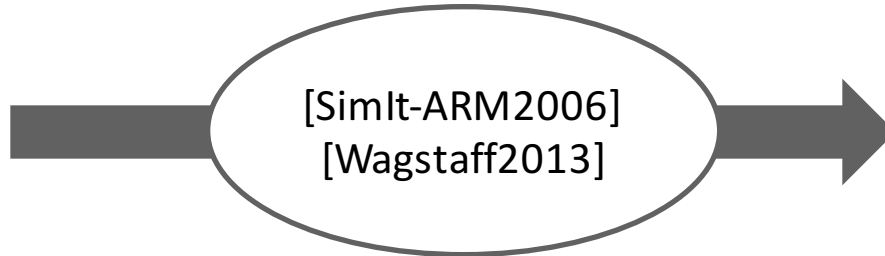


Productivity



Performance

Architectural  
Description  
Language



Instruction Set  
Interpreter in C  
with DBT

**Key Insight:**

Similar productivity-performance challenges for building high-performance interpreters of dynamic languages.  
(e.g. JavaScript, Python)

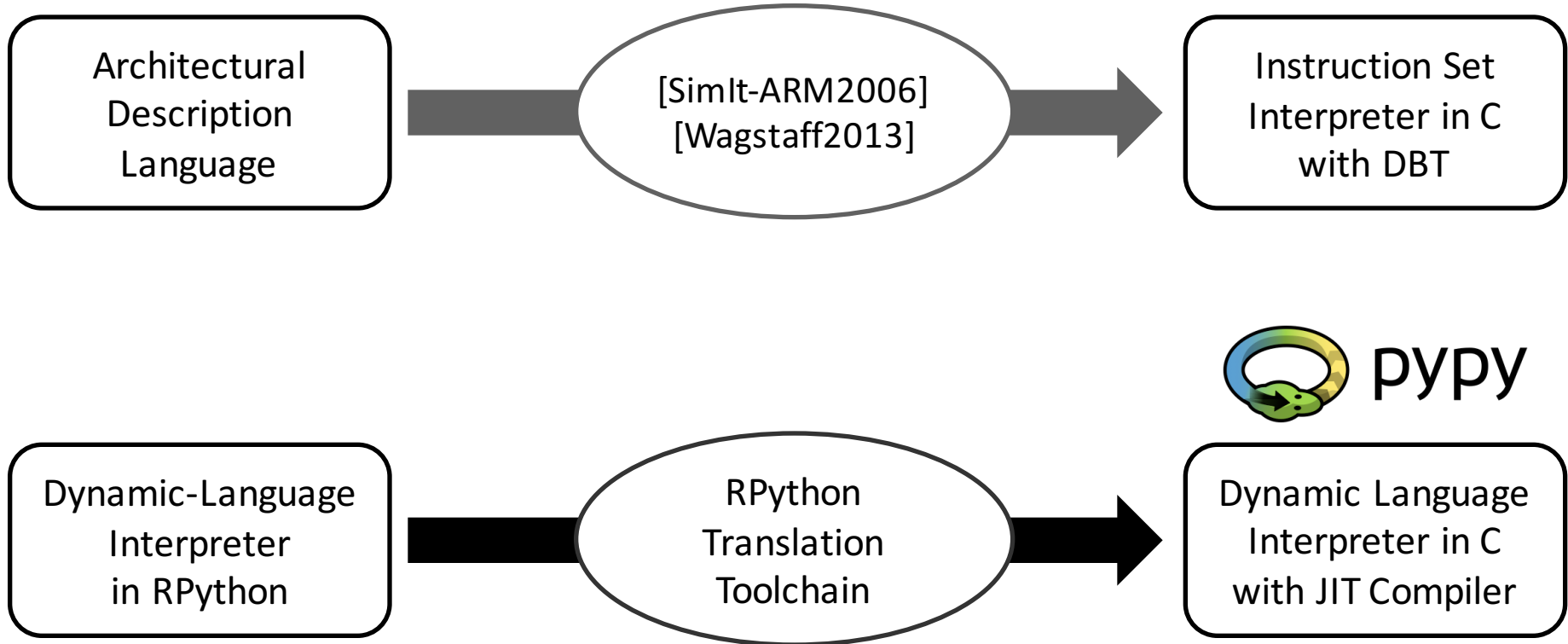


Dynamic Language  
Interpreter in C  
with JIT Compiler

Productivity



Performance



**Meta-Tracing JIT:  
makes JIT generation generic across languages**

Productivity



Performance

Architectural  
Description  
Language



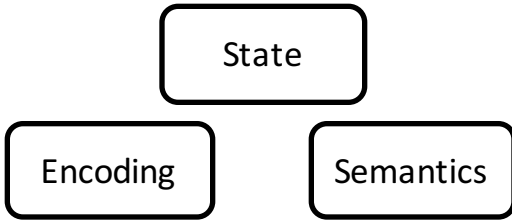
Instruction Set  
Interpreter in C  
with DBT

RPython  
Translation  
Toolchain

**JIT  $\approx$  DBT**

# Pydgin Architecture Description Language

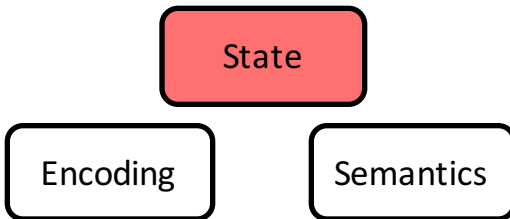
---



**Architectural State**  
**Instruction Encoding**  
**Instruction Semantics**

# Pydgin Architecture Description Language

---



## Architectural State

```
class State( object ):
```

```
def __init__( self, memory, reset_addr=0x400 ):
```

```
    self.pc = reset_addr
```

```
    self.rf = RiscVRegisterFile()
```

```
    self.mem = memory
```

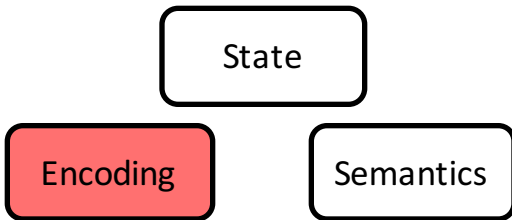
```
# optional state if floating point is enabled
```

```
if ENABLE_FP:
```

```
    self.fp = RiscVFPRegisterFile()
```

```
    self.fcsr = 0
```

# Pydgin Architecture Description Language



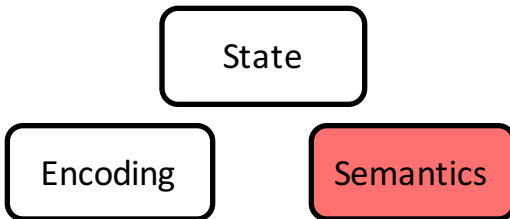
## Instruction Encoding

```
encodings = [  
    # ...  
  
    ['xori', 'xxxxxxxxxxxxxxxxxxxx100xxxxx0010011'],  
    ['ori', 'xxxxxxxxxxxxxxxxxxxx110xxxxx0010011'],  
    ['andi', 'xxxxxxxxxxxxxxxxxxxx111xxxxx0010011'],  
    ['slli', '000000xxxxxxxxxxxx001xxxxx0010011'],  
    ['srli', '000000xxxxxxxxxxxx101xxxxx0010011'],  
    ['srai', '010000xxxxxxxxxxxx101xxxxx0010011'],  
    ['add', '0000000xxxxxxxxxxxx000xxxxx0110011'],  
    ['sub', '0100000xxxxxxxxxxxx000xxxxx0110011'],  
    ['sll', '0000000xxxxxxxxxxxx001xxxxx0110011'],  
  
    # ...  
]
```



# Pydgin Architecture Description Language

---

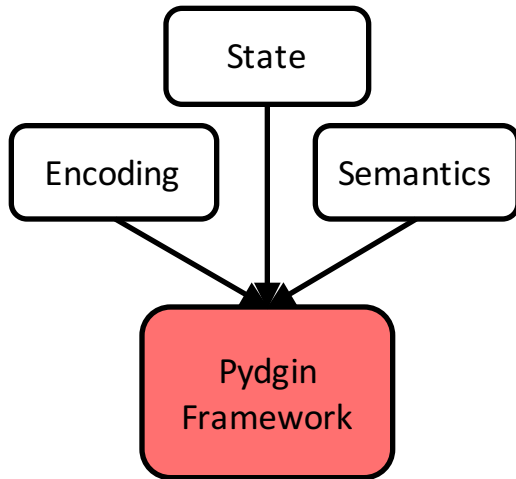


## Instruction Semantics

```
def execute_addi( s, inst ):  
    s.rf[inst.rd] = s.rf[inst.rs1] + inst.i_imm  
    s.pc += 4  
  
def execute_sw( s, inst ):  
    addr = trim_xlen( s.rf[inst.rs1] + inst.s_imm )  
    s.mem.write( addr, 4, trim_32( s.rf[inst.rs2] ) )  
    s.pc += 4  
  
def execute_beq( s, inst ):  
    if s.rf[inst.rs1] == s.rf[inst.rs2]:  
        s.pc = trim_xlen( s.pc + inst.sb_imm )  
    else:  
        s.pc += 4
```

# Pydgin Framework

---



## Interpreter Loop

```
def instruction_set_interpreter( memory ):
    state = State( memory )

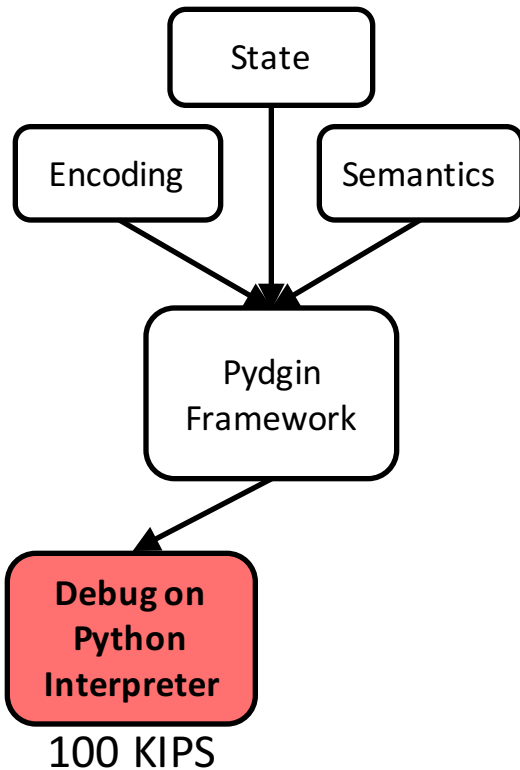
    while True:

        pc      = state.fetch_pc()

        inst    = memory[ pc ]      # fetch
        execute = decode( inst )   # decode
        execute( state, inst )     # execute
```

# Pydgin Framework

---



## Interpreter Loop

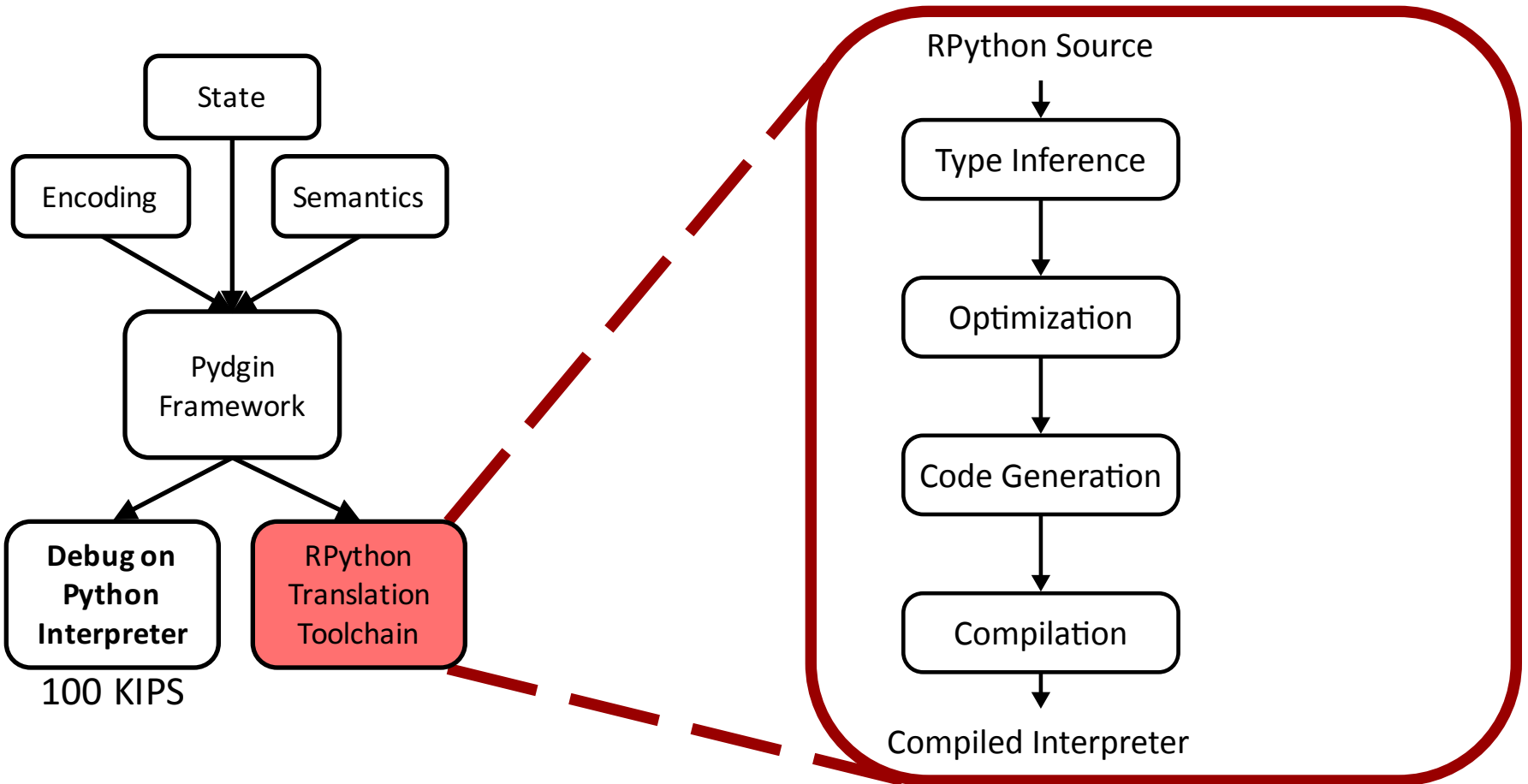
```
def instruction_set_interpreter( memory ):
    state = State( memory )

    while True:

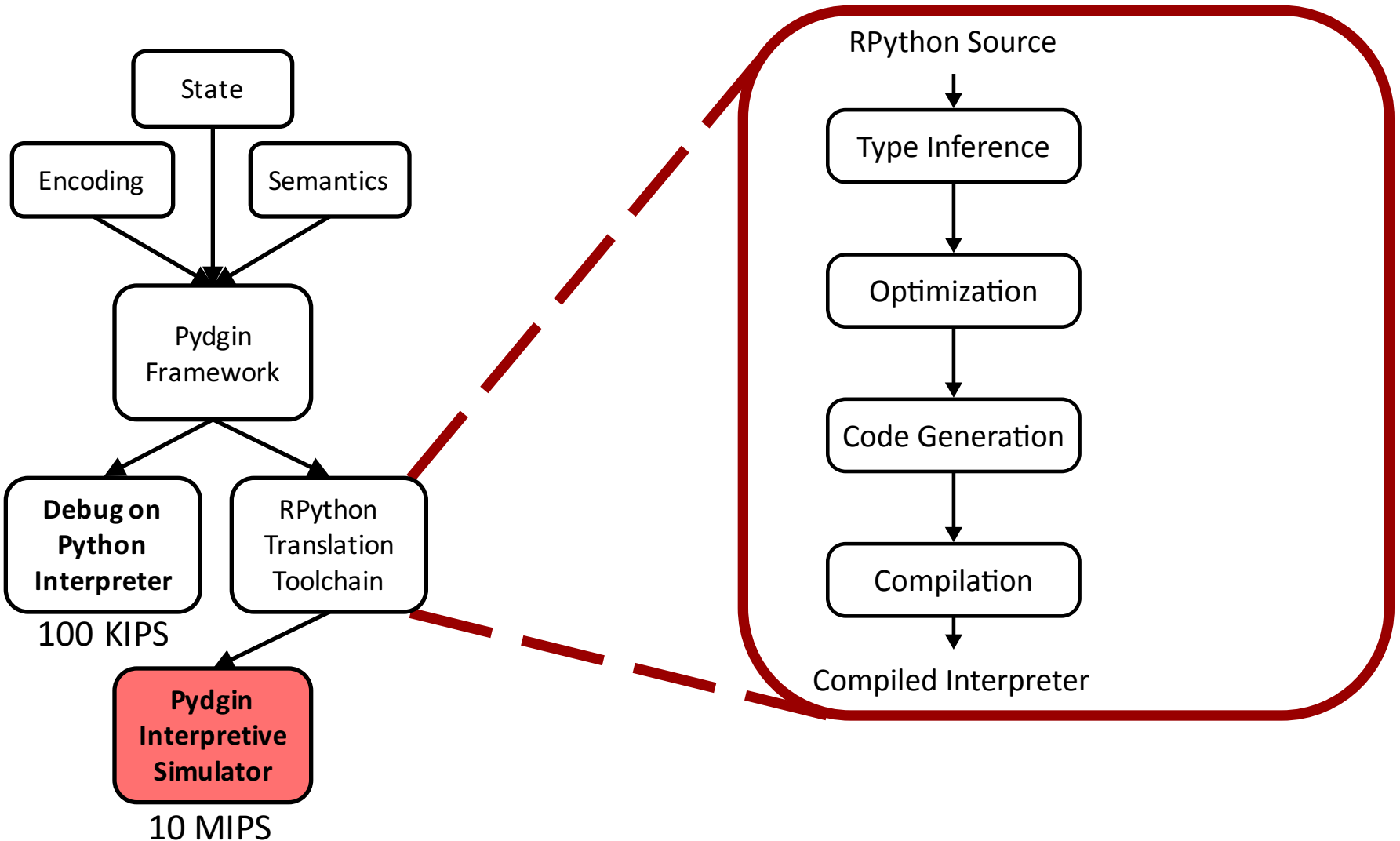
        pc      = state.fetch_pc()

        inst    = memory[ pc ]      # fetch
        execute = decode( inst )   # decode
        execute( state, inst )     # execute
```

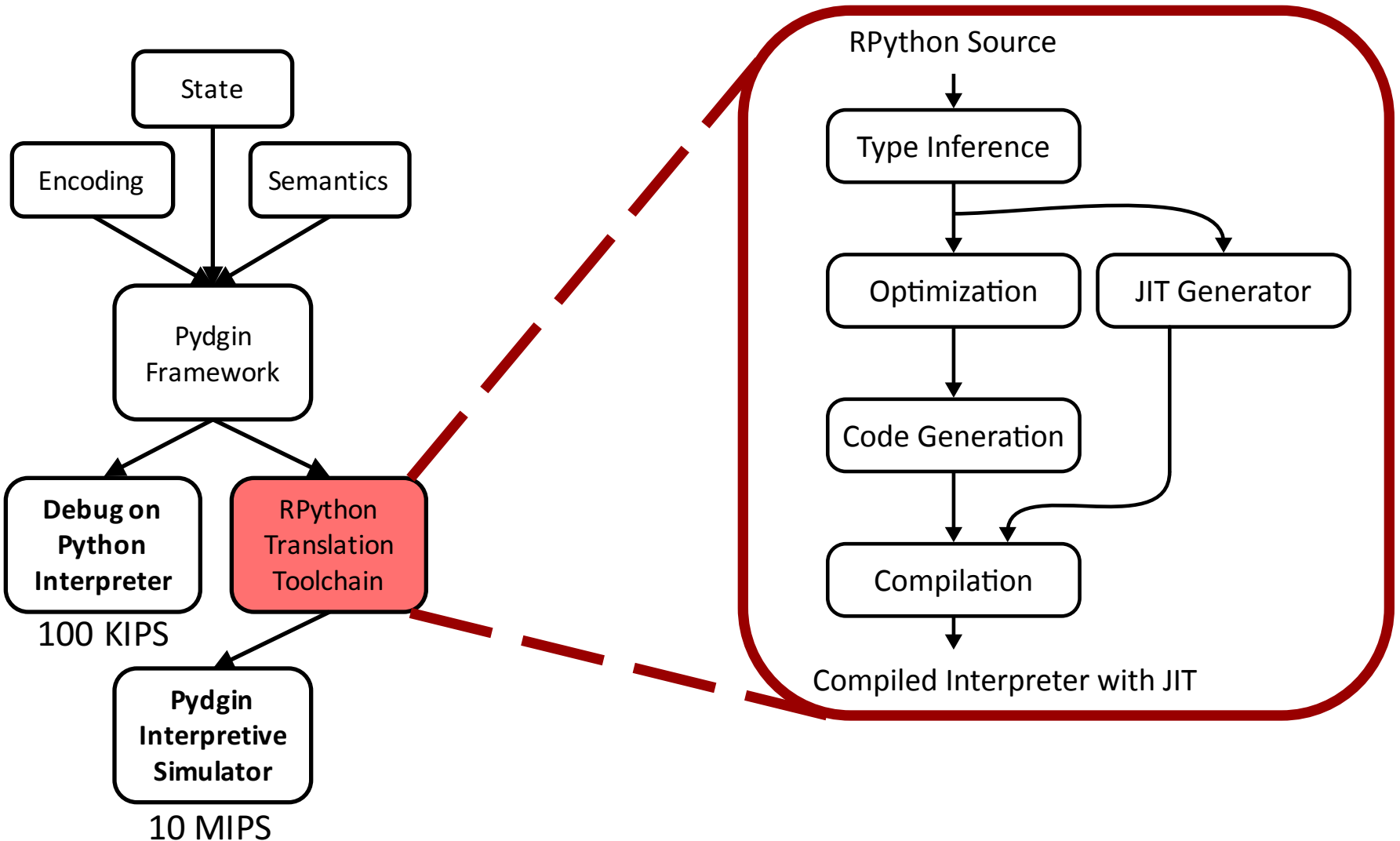
# The RPython Translation Toolchain



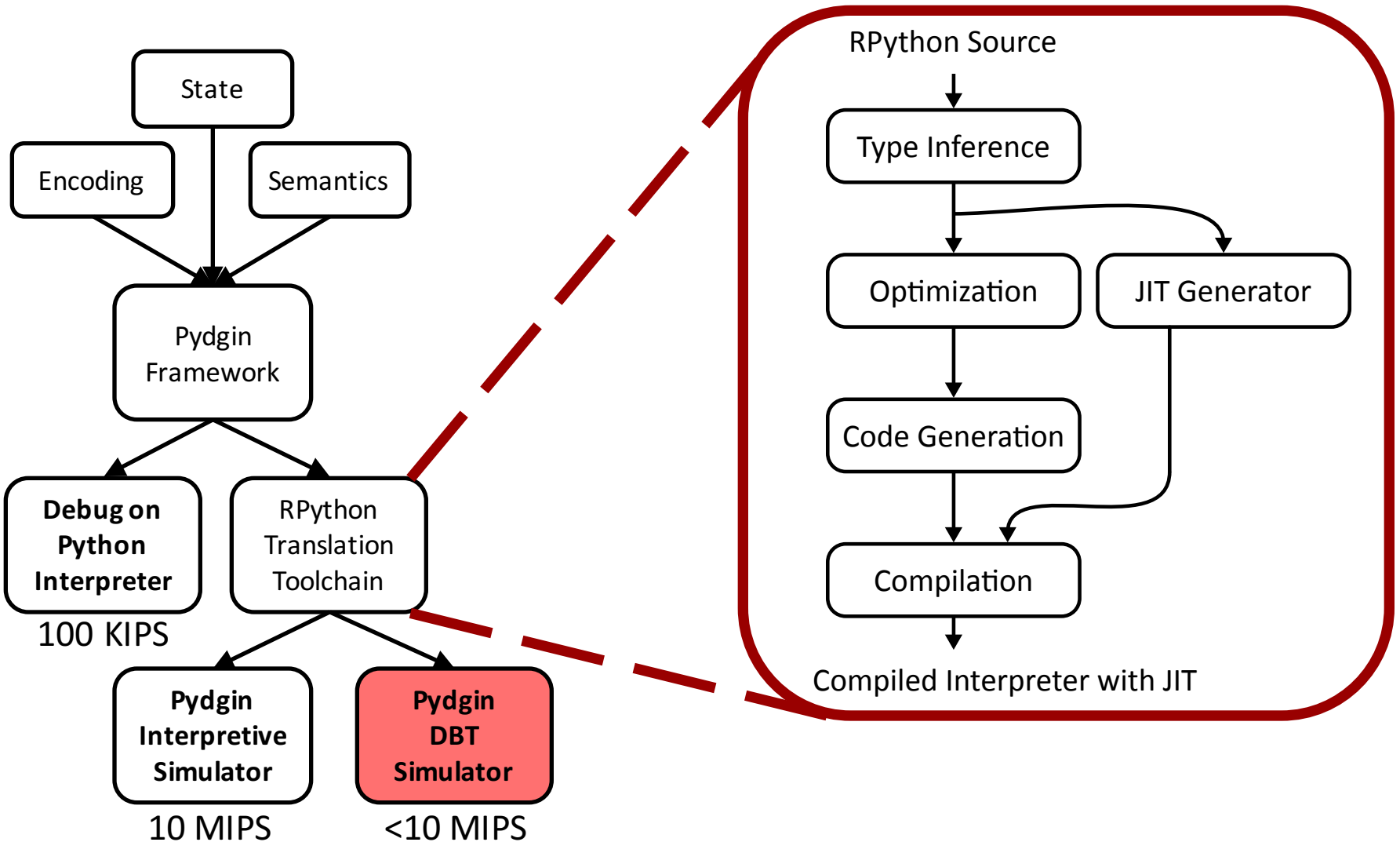
# The RPython Translation Toolchain



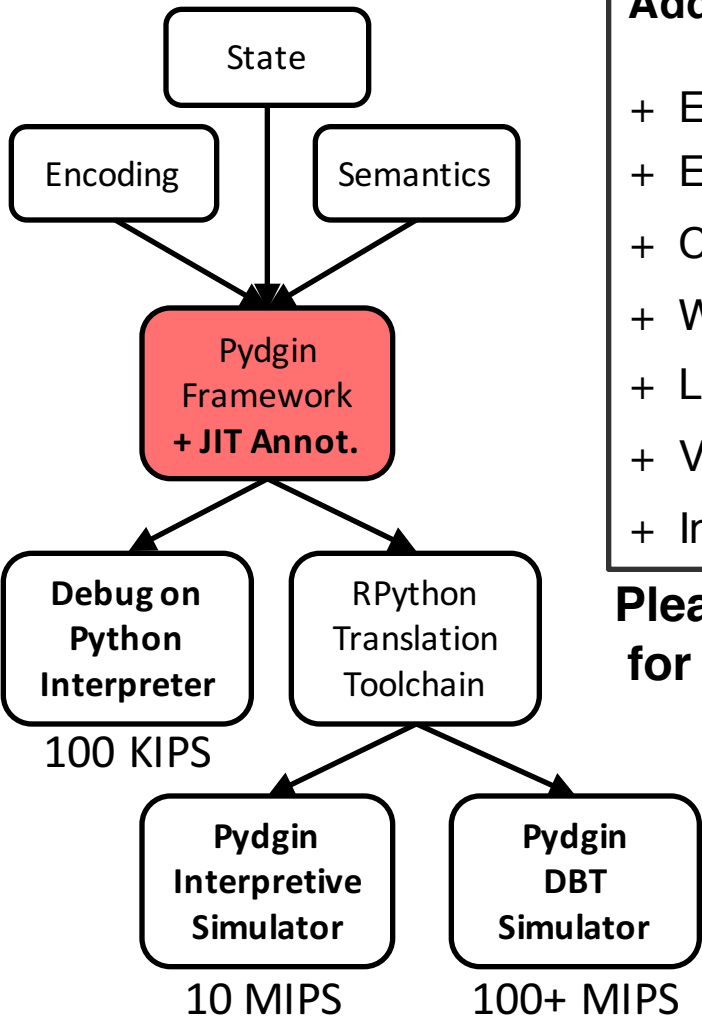
# The RPython Translation Toolchain



# The RPython Translation Toolchain



# JIT Annotations and Optimizations

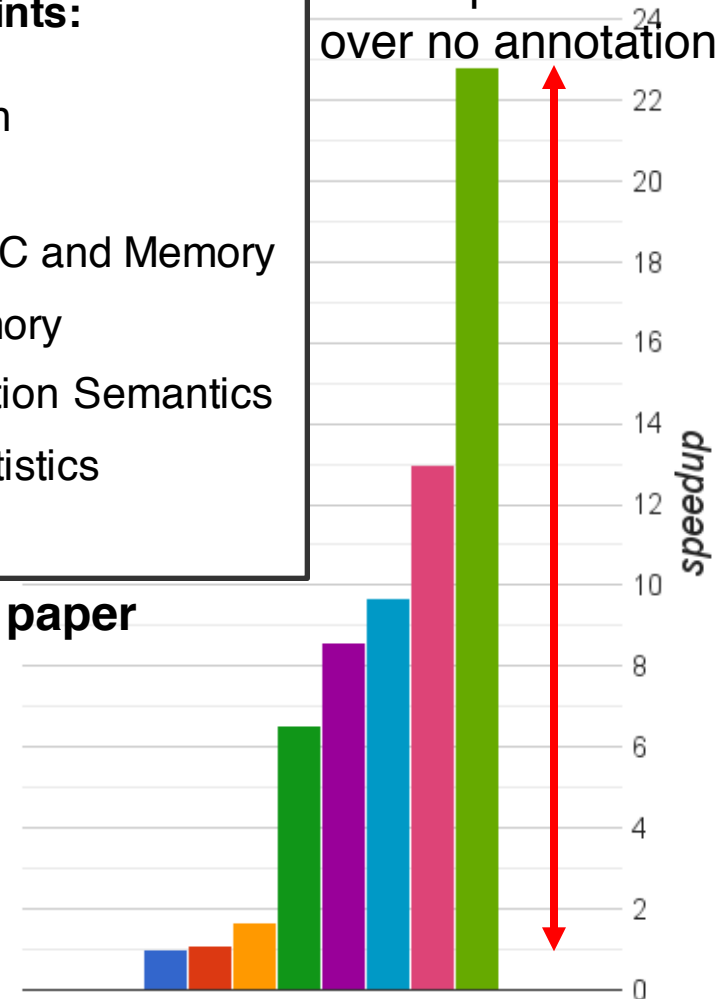


## Additional RPython JIT hints:

- + Elidable Instruction Fetch
- + Elidable Decode
- + Constant Promotion of PC and Memory
- + Word-Based Target Memory
- + Loop Unrolling in Instruction Semantics
- + Virtualizable PC and Statistics
- + Increased Trace Limit

**Please see our ISPASS paper for more details!**

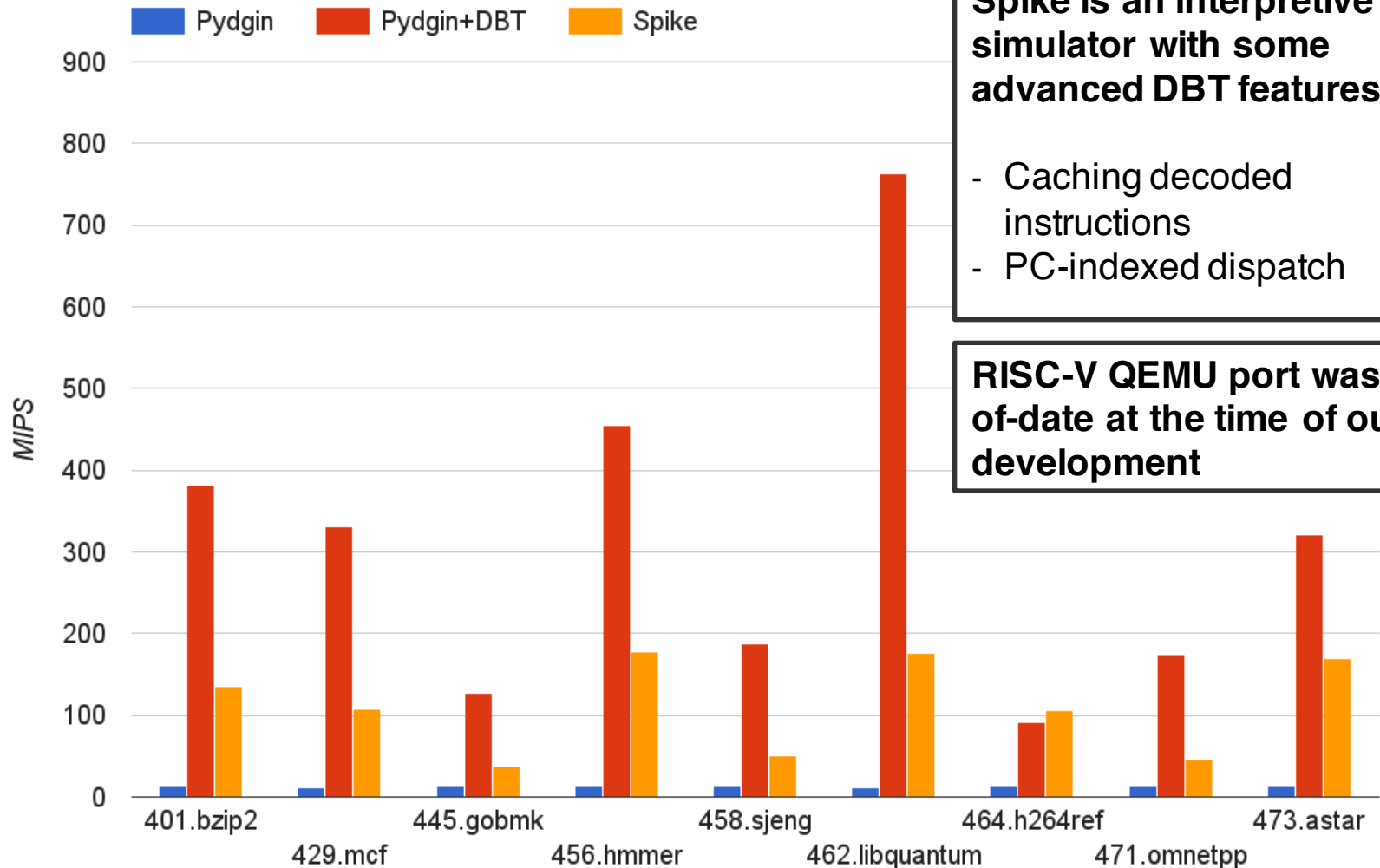
23X improvement over no annotations



**SPECINT2006 on ARM**



# Pydgin Performance



**Spike is an interpretive simulator with some advanced DBT features:**

- Caching decoded instructions
- PC-indexed dispatch

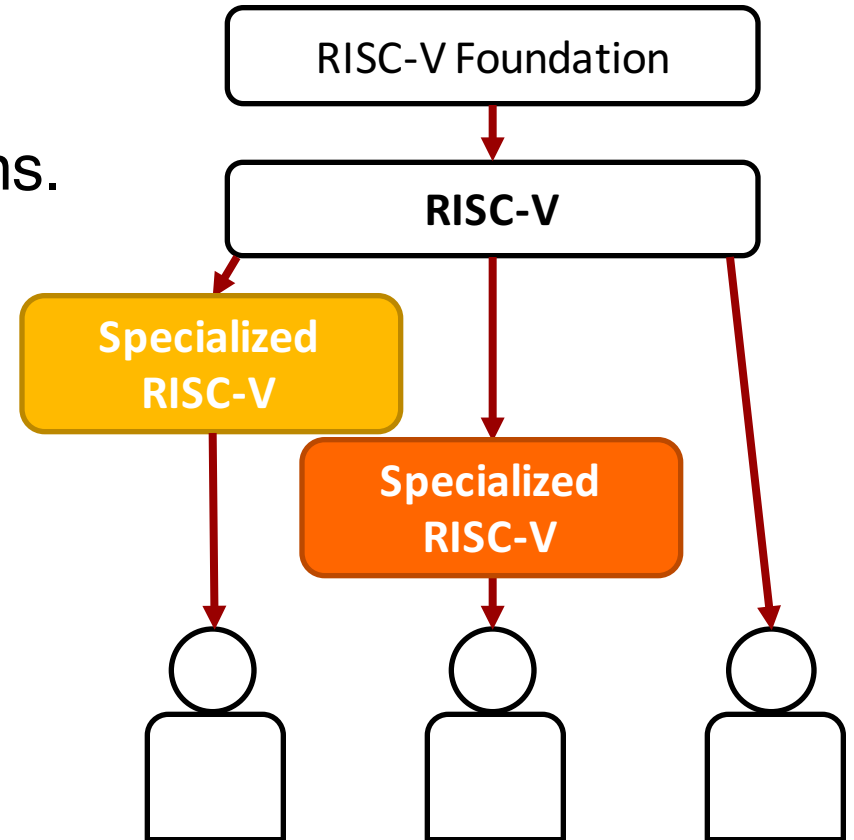
**RISC-V QEMU port was out-of-date at the time of our development**

# Pydgin Productivity

---

RISC-V encourages ISA extensions.

- Productive Development
- Productive Extensibility
- Productive Instrumentation



# Pydgin RISC-V Development

```
2015-06-15 Derek Lockhart [riscv] add isa.py skeleton
2015-06-15 Derek Lockhart [riscv] add riscv-sim.py toplevel
2015-06-15 Berkin Ibeyi [riscv] add machine.py
2015-06-15 Derek Lockhart [riscv] ?s to xs, alignment
2015-06-15 Derek Lockhart [riscv] lowercase inst names, add placeholder funcs
2015-06-15 Berkin Ibeyi [riscv] first attempt at instruction.py
2015-06-15 Derek Lockhart [riscv] add helpers.py, some hacky inst impls
2015-06-15 Derek Lockhart [riscv] addi inst
2015-06-15 Berkin Ibeyi [riscv] fix syntax errors, add bootstrap
2015-06-15 Derek Lockhart [riscv] add xori, ori, xor, or
2015-06-15 Derek Lockhart [riscv] beq, bne, blt, bge, bltu, bgeu
2015-06-15 Derek Lockhart [riscv] add hacky, temp impl of csrwr
2015-06-15 Derek Lockhart [riscv] fix typos in branch insts, hacky impl of jal, jalr
2015-06-15 Berkin Ibeyi [pydgin] add elf64 reader
2015-06-15 Berkin Ibeyi [riscv] remove parc junk
2015-06-15 Berkin Ibeyi [riscv] fix reset vector
2015-06-15 Berkin Ibeyi [riscv] fixes and typos in semantics etc
2015-06-15 Berkin Ibeyi [pydgin] remove debug prints
2015-06-15 Derek Lockhart [riscv] slti, sltiu, andi, sub, slt, sltu instructions
2015-06-15 Derek Lockhart [riscv] fix forgotten rf reads, cleanup
2015-06-15 Berkin Ibeyi [riscv] add sext_32
2015-06-15 Derek Lockhart [riscv] more forgotten rf read fixes
2015-06-15 Derek Lockhart [riscv] auipc, srl, srai, sll, srl, sra, sllw, srlw, srarw
2015-06-15 Derek Lockhart [riscv] addw, sllw, srlw, srarw
2015-06-15 Berkin Ibeyi [pydgin] rf printing can be done for 64bits
2015-06-15 Berkin Ibeyi [riscv] use sext instead of signed for inst fields
2015-06-15 Berkin Ibeyi [riscv] have rf automatically trim the write vals
2015-06-15 Berkin Ibeyi [riscv] fix typo
2015-06-15 Berkin Ibeyi [riscv] fix signed in isa
2015-06-15 Derek Lockhart [riscv] fix branch/jump pc, csrwr pass/fail, fence noop
2015-06-15 Yunsup Lee [riscv] fix csrwr
```

```
2015-06-15 Derek Lockhart [riscv] add isa.py skeleton
2015-06-15 Derek Lockhart [riscv] add riscv-sim.py toplevel
2015-06-15 Berkin Ibeyi [riscv] add machine.py
2015-06-15 Derek Lockhart [riscv] ?s to xs, alignment
2015-06-15 Derek Lockhart [riscv] lowercase inst names, add placeholder funcs
2015-06-15 Berkin Ibeyi [riscv] first attempt at instruction.py
2015-06-15 Derek Lockhart [riscv] add helpers.py, some hacky inst impls
2015-06-15 Derek Lockhart [riscv] addi inst
2015-06-15 Berkin Ibeyi [riscv] fix syntax errors, add bootstrap
2015-06-15 Derek Lockhart [riscv] add xori, ori, xor, or
2015-06-15 Derek Lockhart [riscv] beq, bne, blt, bge, bltu, bgeu
2015-06-15 Derek Lockhart [riscv] add hacky, temp impl of csrwr
2015-06-15 Derek Lockhart [riscv] fix typos in branch insts, hacky impl of jal, jalr
2015-06-15 Berkin Ibeyi [pydgin] add elf64 reader
2015-06-15 Berkin Ibeyi [riscv] remove parc junk
2015-06-15 Berkin Ibeyi [riscv] fix reset vector
2015-06-15 Berkin Ibeyi [riscv] fixes and typos in semantics etc
2015-06-15 Berkin Ibeyi [pydgin] remove debug prints
2015-06-15 Derek Lockhart [riscv] slti, sltiu, andi, sub, slt, sltu instructions
2015-06-15 Derek Lockhart [riscv] fix forgotten rf reads, cleanup
2015-06-15 Berkin Ibeyi [riscv] add sext_32
2015-06-15 Derek Lockhart [riscv] more forgotten rf read fixes
2015-06-15 Derek Lockhart [riscv] auipc, srl, srai, sll, srl, sra, sllw, srlw, srarw
2015-06-15 Berkin Ibeyi [pydgin] rf printing can be done for 64bits
2015-06-15 Berkin Ibeyi [riscv] use sext instead of signed for inst fields
2015-06-15 Berkin Ibeyi [riscv] have rf automatically trim the write vals
2015-06-15 Berkin Ibeyi [riscv] fix typo
2015-06-15 Berkin Ibeyi [riscv] fix signed in isa
2015-06-15 Derek Lockhart [riscv] fix branch/jump pc, csrwr pass/fail, fence noop
2015-06-15 Yunsup Lee [riscv] fix csrwr
2015-06-15 Berkin Ibeyi [riscv] fix csrwr
2015-06-15 Berkin Ibeyi [riscv] fix csrwr
2015-06-17 Derek Lockhart [riscv] fence_i and amov_w instructions
2015-06-17 Berkin Ibeyi [pydgin,parc] add many typing annotations for 32-bit machines
2015-06-17 Berkin Ibeyi [pydgin,pydgin] more typing annots for this to work on 64bit
2015-06-17 Berkin Ibeyi [pydgin,parc] make this runnable in interpretive mode
2015-06-17 Berkin Ibeyi [pydgin] more changes to get it working on 64bit
2015-06-17 Berkin Ibeyi [pydgin] clean up comments
2015-06-17 Berkin Ibeyi [riscv] rpython type hacks to get translated
2015-06-17 Berkin Ibeyi [riscv] can run tests on the translated pydgin, beautify test outs
2015-06-17 Berkin Ibeyi [riscv] add gitignore
2015-06-17 Berkin Ibeyi [riscv] more rpython type hacks for trim
2015-06-18 Berkin Ibeyi [riscv] declare virtualizables to keep rpython happy
2015-06-18 Berkin Ibeyi [riscv] do some arith magic to calculate mulh instructions
2015-06-18 Berkin Ibeyi [riscv] some usability improvements to test-all riscv
2015-06-19 Berkin Ibeyi [riscv] bootstrap differently for non-tests
2015-06-19 Derek Lockhart [riscv] add floating-point register file, cleanup Riscv RF init
2015-06-19 Derek Lockhart [riscv] add xlen/flen state, also csr and fcsr reg state
2015-06-19 Berkin Ibeyi [riscv] impl syscall emul wrappers
2015-06-20 Derek Lockhart [riscv] fix type in name of fcsr register
2015-06-20 Derek Lockhart [riscv] implement fp loads, mvs, add/sub/mul using softfloat
2015-06-20 Derek Lockhart [riscv] modify test_all script to support exec from build dir
2015-06-20 Derek Lockhart [riscv] add fclass_s, fclass_d, fmv_d_x insns
2015-06-20 Derek Lockhart [riscv] add f(le,lt,eq)_s,d insns
2015-06-20 Derek Lockhart [riscv] add fcv*_w instructions
2015-06-20 Derek Lockhart [softfloat] add missing .h files, oops
2015-06-20 Derek Lockhart [riscv] fix bugs in fp condition flags, flw
2015-06-20 Berkin Ibeyi [riscv] ignore fst, fld insns
2015-06-20 Berkin Ibeyi [riscv] override default stat struct with riscv-specific struct
2015-06-20 Berkin Ibeyi [riscv] add regmap
2015-06-20 Berkin Ibeyi [riscv] hack in machine to get translated
2015-06-20 Berkin Ibeyi [riscv] impl syscall_init which sets up the stack
2015-06-20 Berkin Ibeyi [riscv] "riscv" hack in machine to get translated"
2015-06-20 Derek Lockhart [riscv] add fcv*_w instructions (details)
2015-06-20 Derek Lockhart [riscv] major refactor to modularize inst defs, disable fp
2015-06-21 Derek Lockhart [riscv] add fmv_s_d, fmv_d_s, fmv_d_d insns
2015-06-21 Derek Lockhart [riscv] add fmaadd, fmaddd, fmsub, fmsubd .s and .d insns
2015-06-21 Berkin Ibeyi [riscv] don't import fp if fp not enabled
2015-06-21 Derek Lockhart [riscv] hacky impl of fmin,max)_s,d, doesn't support NaN correctly
2015-06-21 Berkin Ibeyi [riscv] minor fixes to allow translation w/o fp
2015-06-21 Derek Lockhart [riscv] impl fsgnj, fsgnjn, fsgnjx insns for .s and .d
2015-06-21 Derek Lockhart [riscv] impl fsw and fsd instructions
2015-06-21 Derek Lockhart [riscv] impl lr_w and sc_w insns; all asm tests pass!
2015-06-21 Derek Lockhart [riscv] impl lr_d and sc_d insns, untested
2015-06-21 Berkin Ibeyi [riscv] fix ld/st/jump bugs that show up in dense mem
2015-06-21 Berkin Ibeyi [riscv] trim_64 fp ld/st ops as well
2015-06-21 Berkin Ibeyi [riscv] map fsd and fld to nop if fp not enabled
2015-06-21 Berkin Ibeyi [riscv] start the stack from end of memory
2015-06-21 Berkin Ibeyi [riscv] order tests
2015-06-21 Berkin Ibeyi [riscv] grow mem for some spec bmarks
2015-06-21 Berkin Ibeyi [scripts] added riscv as a translation target
2015-06-22 Derek Lockhart [riscv] new interface to softfloat, cffi works, widening errs on rffi
2015-06-22 Derek Lockhart [riscv] horrible hack for register file issue
2015-06-22 Derek Lockhart [riscv,softfloat] type hacks to enable translation
2015-06-22 Derek Lockhart [riscv] cleanup of hacky FP reg file
2015-06-22 Derek Lockhart [softfloat,scripts] hacky ./so-dylib workaround for rffi bug
2015-06-22 Derek Lockhart [riscv] enable FP by default since it translates now
2015-06-23 Derek Lockhart [scripts] fix bug in softfloat script
2015-06-23 Derek Lockhart [riscv] add type annotation to fix rpython transl problem
2015-06-23 Derek Lockhart [scripts] fix hacky script to find lib paths correctly
2015-07-14 Berkin Ibeyi [riscv] unsigned annotation to keep rpython happy
2015-07-14 Berkin Ibeyi [pydgin,riscv] use custom not impl inst error for better err reporting
2015-07-15 Berkin Ibeyi [riscv] impl csrwr and csrwr to access fcsr
2015-07-15 Berkin Ibeyi [scripts] make building work in build dir
```



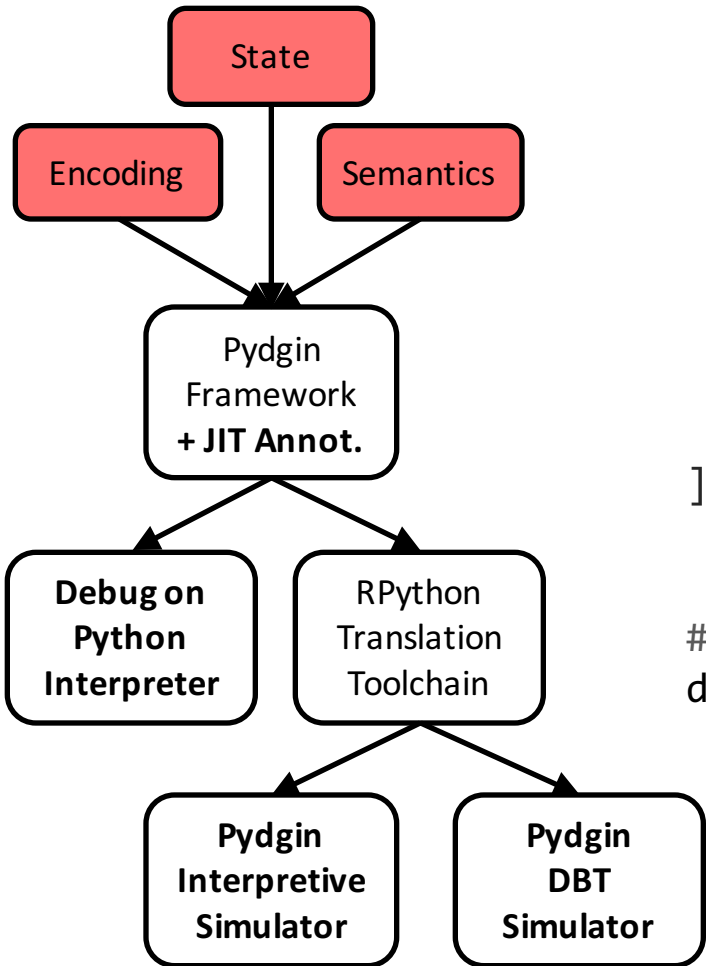
# Pydgin RISC-V Development

100+ MIPS simulator  
after 9 days of development!

```
2013-06-15 Derek Lockhart [riscv] add isa.py skeleton
2013-06-15 Derek Lockhart [riscv] add riscv-sim.py toplevel
2013-06-15 Berkin Ibeyi [riscv] add machine.py
2013-06-15 Derek Lockhart [riscv] %s to %s, alignment
2013-06-15 Derek Lockhart [riscv] lowercase inst names, add placeholder funcs
2013-06-15 Derek Lockhart [riscv] first attempt at instruction.py
2013-06-15 Derek Lockhart [riscv] add helpers.py, some hacky inst impls
2013-06-15 Derek Lockhart [riscv] addw inst
2013-06-15 Berkin Ibeyi [riscv] fix syntax errors, add bootstrap
2013-06-15 Derek Lockhart [riscv] add mori, ori, nor, or
2013-06-15 Derek Lockhart [riscv] beq, bne, blt, bge, bltu, bgeu
2013-06-15 Derek Lockhart [riscv] add hacky, temp impl of csrwr
2013-06-15 Derek Lockhart [riscv] fix typos in branch insts, hacky impl of jal, jalr
2013-06-15 Berkin Ibeyi [pydgin] add elf64 reader
2013-06-15 Berkin Ibeyi [riscv] remove parc junk
2013-06-15 Berkin Ibeyi [riscv] fix reset vector
2013-06-15 Berkin Ibeyi [riscv] fixes and typos in semantics etc
2013-06-15 Berkin Ibeyi [pydgin] remove debug prints
2013-06-15 Derek Lockhart [riscv] sllr, sllw, andi, sub, sll, sltu instructions
2013-06-15 Derek Lockhart [riscv] fix forgotten rf reads, cleanup
2013-06-15 Berkin Ibeyi [riscv] add sext_32
2013-06-15 Derek Lockhart [riscv] more forgotten rf read fixes
2013-06-15 Derek Lockhart [riscv] auspc, srl, srai, sll, srl, sra, sllw, srlw, srw
2013-06-15 Derek Lockhart [riscv] addw, sllw, srlw, srw
2013-06-15 Berkin Ibeyi [pydgin] rf printing can be done for 64bits
2013-06-15 Berkin Ibeyi [riscv] use sext instead of signed for inst fields
2013-06-15 Berkin Ibeyi [riscv] have rf automatically trim the write vals
2013-06-15 Berkin Ibeyi [riscv] fix type
2013-06-15 Berkin Ibeyi [riscv] fix signed in isa
2013-06-15 Derek Lockhart [riscv] fix branch/jump pc, csrwr pass/fail, fence noop
2013-06-15 Yunsup Lee [riscv] fix csrwr
2013-06-16 Berkin Ibeyi [riscv] use FatalError instead of Exception
2013-06-16 Berkin Ibeyi [riscv] fix l_imm, add variable len sext
2013-06-16 Berkin Ibeyi [riscv] zero upper bits if not signed in sext_32
2013-06-16 Yunsup Lee [riscv] all 0's is not a nop; it's illegal
2013-06-16 Berkin Ibeyi [riscv] sext_32_imm correctly
2013-06-16 Yunsup Lee [riscv] fix andi
2013-06-16 Yunsup Lee [riscv] fix ori/xori as well
2013-06-16 Yunsup Lee [riscv] fix some typos
2013-06-16 Berkin Ibeyi [riscv] fix sra
2013-06-16 Berkin Ibeyi [riscv] add script to test all
2013-06-16 Berkin Ibeyi [riscv] print test summary
2013-06-16 Berkin Ibeyi [riscv] fix srsw
2013-06-16 Berkin Ibeyi [riscv] fix srw
2013-06-16 Berkin Ibeyi [riscv] fix srlw, srlw
2013-06-16 Berkin Ibeyi [pydgin] sparse mem prints debug
2013-06-16 Berkin Ibeyi [riscv] impl lw
2013-06-16 Berkin Ibeyi [riscv] impl lb, lh, lbu, lhu, lwu
2013-06-16 Berkin Ibeyi [riscv] impl sb, sh
2013-06-16 Berkin Ibeyi [riscv] impl ld, sd
2013-06-17 Berkin Ibeyi [riscv] impl mulw, divw, remw insts
2013-06-17 Derek Lockhart [riscv] fence_i and amow_w instructions
2013-06-17 Berkin Ibeyi [pydgin,parc] add many typing annotations for 32-bit machines
2013-06-17 Berkin Ibeyi [pydgin] some more annotations to allow translation w/ debug
2013-06-17 Berkin Ibeyi [parc,pydgin] more typing annots for this to work on 64bit
2013-06-17 Berkin Ibeyi [pydgin,parc] make this runnable in interpretive mode
2013-06-17 Berkin Ibeyi [pydgin] more changes to get it working on 64bit
2013-06-17 Berkin Ibeyi [pydgin] clean up comments
2013-06-17 Berkin Ibeyi [riscv] rpython type hacks to get translated
2013-06-17 Berkin Ibeyi [riscv] can run tests on the translated pydgin, beautify test outs
2013-06-17 Berkin Ibeyi [riscv] add gittignore
2013-06-17 Berkin Ibeyi [riscv] more rpython type hacks for trim
2013-06-18 Berkin Ibeyi [riscv] declare virtualizables to keep rpython happy
2013-06-18 Berkin Ibeyi [riscv] do some arith magic to calculate mulhw instructions
2013-06-18 Berkin Ibeyi [riscv] some usability improvements to test-all script
2013-06-19 Berkin Ibeyi [riscv] bootstrap differently for non-tests
2013-06-19 Derek Lockhart [riscv] add floating-point register file, cleanup RiscV RF init
2013-06-19 Derek Lockhart [riscv] add xlen/flen state, also csr and fcsr reg state
2013-06-19 Berkin Ibeyi [riscv] impl syscall emul wrappers
2013-06-19 Berkin Ibeyi [syscall-tests] compile syscall tests for riscv
2013-06-20 Derek Lockhart [softfloat,scripts] add softfloat source = build script
2013-06-20 Derek Lockhart [softfloat,scripts] add error checking, import f(32,64)_(add,sub,mul)
2013-06-20 Derek Lockhart [riscv] fix type in name of fcsr register
2013-06-20 Derek Lockhart [riscv] implement fp loads, mvs, add/sub/mul using softfloat
2013-06-20 Derek Lockhart [riscv] modify test-all script to support exec from build dir
2013-06-20 Derek Lockhart [riscv] add fclass_s, fclass_d, fmv_d_x insns
2013-06-20 Derek Lockhart [riscv] add f(le,lt,eq)_s,d insns
2013-06-20 Derek Lockhart [riscv] add fcv*_w instructions
2013-06-20 Derek Lockhart [softfloat] add missing .i files, oops
2013-06-20 Derek Lockhart [riscv] fix bugs in fp condition flags, flw
2013-06-20 Berkin Ibeyi [riscv] ignore fst, fld insts
2013-06-20 Berkin Ibeyi [riscv] override default stat struct with riscv-specific struct
2013-06-20 Berkin Ibeyi [riscv] add regmap
2013-06-20 Berkin Ibeyi [riscv] hack in machine to get translated
2013-06-20 Berkin Ibeyi [riscv] impl syscall_init which sets up the stack
2013-06-20 Berkin Ibeyi [riscv] hack in machine to get translated
2013-06-20 Berkin Ibeyi [riscv] add fcv*_w instructions (details)
2013-06-20 Derek Lockhart [riscv] major refactor to modularize inst defs, disable fp
2013-06-21 Derek Lockhart [riscv] add fmv_s, fmv_d, fmv*_s, fmv*_d insts
2013-06-21 Derek Lockhart [riscv] add fmaadd, fmaddd, fmsub, fmsub*_s and .d insts
2013-06-21 Berkin Ibeyi [riscv] don't import fp if fp not enabled
2013-06-21 Derek Lockhart [riscv] hacky impl of fmin,max)_s,d, doesn't support NaN correctly
2013-06-21 Berkin Ibeyi [riscv] minor fixes to allow translation w/o fp
2013-06-21 Derek Lockhart [riscv] impl fsgnj, fsgnjn, fsgnjx insns for .s and .d
2013-06-21 Derek Lockhart [riscv] impl fsw and fsd instructions
2013-06-21 Derek Lockhart [riscv] impl lr_w and sc_w insts; all asm tests pass!
2013-06-21 Derek Lockhart [riscv] impl lr_d and sc_d insts, untested
2013-06-21 Berkin Ibeyi [riscv] fix ld/st/jump bugs that show up in dense mem
2013-06-21 Berkin Ibeyi [riscv] trim_64 fp ld/st ops as well
2013-06-21 Berkin Ibeyi [riscv] map fsd and fld to nop if fp not enabled
2013-06-21 Berkin Ibeyi [riscv] start the stack from end of memory
2013-06-21 Berkin Ibeyi [riscv] order tests
2013-06-21 Berkin Ibeyi [riscv] grow mem for some spec bmarks
2013-06-21 Berkin Ibeyi [scripts] added riscv as a translation target
2013-06-22 Derek Lockhart [riscv] new interface to softfloat, cffi works, widening errs on rffi
2013-06-22 Derek Lockhart [riscv] horrible hack for register file issue
2013-06-22 Derek Lockhart [riscv,softfloat] type hacks to enable translation
2013-06-22 Derek Lockhart [riscv] cleanup of hacky FP reg file
2013-06-22 Derek Lockhart [softfloat,scripts] hacky ./so-dylib workaround for rffi bug
2013-06-22 Derek Lockhart [riscv] enable FP by default since it translates now
2013-06-23 Derek Lockhart [scripts] fix bug in softfloat script
2013-06-23 Derek Lockhart [riscv] add type annotation to fix rpython transl problem
2013-06-23 Derek Lockhart [scripts] fix hacky script to find lib paths correctly
2013-07-14 Berkin Ibeyi [riscv] unsigned annotation to keep rpython happy
2013-07-14 Berkin Ibeyi [pydgin,riscv] use custom not impl inst error for better err reporting
2013-07-15 Berkin Ibeyi [riscv] impl csrr and csrrw to access fcsr
2013-07-15 Berkin Ibeyi [scripts] make building work in build dir
```



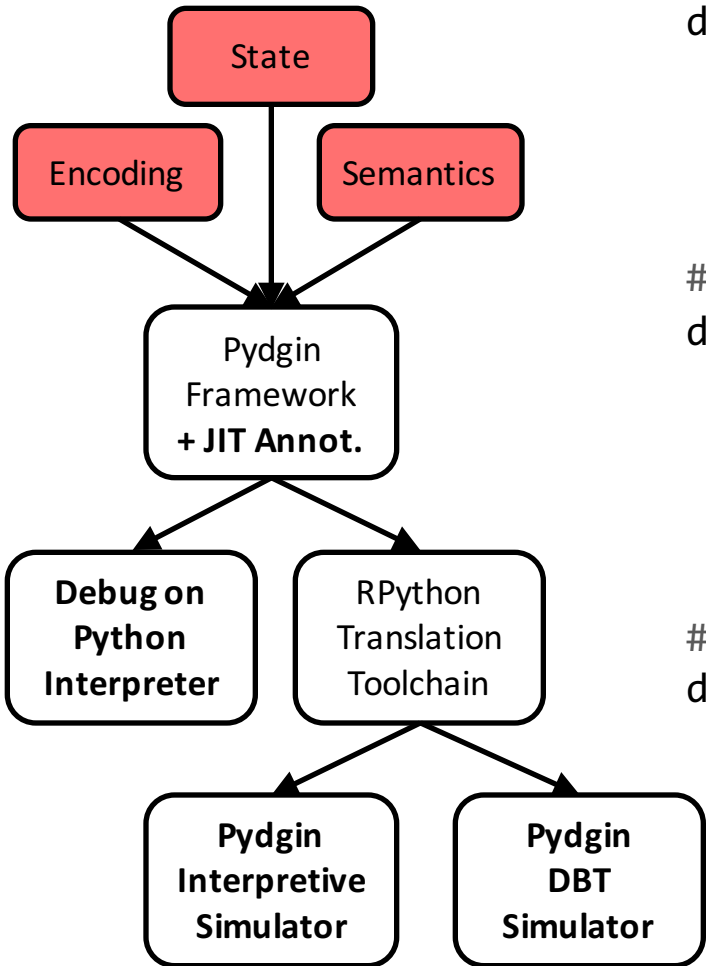
# Pydgin Extensibility



```
encodings = [  
    # ...  
    ['andi',      'xxxxxxxxxxxxxxxxxxxx111xxxxx0010011'],  
    ['slli',      '000000xxxxxxxxxxxxx001xxxxx0010011'],  
    ['srli',      '000000xxxxxxxxxxxxx101xxxxx0010011'],  
    ['srai',      '010000xxxxxxxxxxxxx101xxxxx0010011'],  
    ['add',       '0000000xxxxxxxxxxxxx000xxxxx0110011'],  
    # ...  
    ['gcd',       'xxxxxxxxxxxxxxxxxxxx000xxxxx1011011'],  
    # ...  
]
```

```
# greatest common divisor semantics  
def execute_gcd( s, inst ):  
    a, b = s.rf[inst.rs1], s.rf[inst.rs2]  
    while b:  
        a, b = b, a%b  
        s.rf[inst.rd] = a  
        s.pc += 4
```

# Pydgin Instrumentation



```
# count number of adds
def execute_addi( s, inst ):
    s.rf[inst.rd] = s.rf[inst.rs1] + inst.i_imm
    s.num_adds += 1
    s.pc += 4

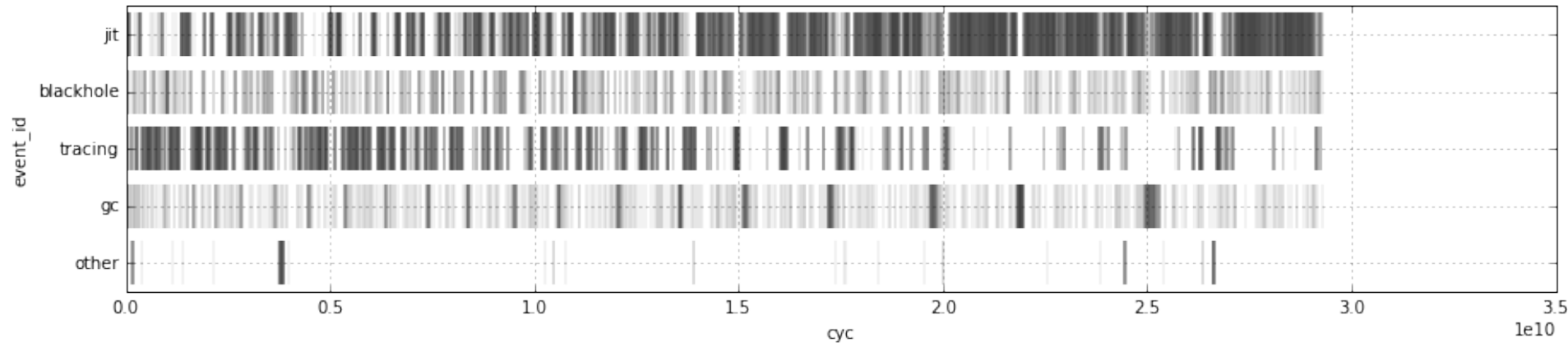
# count misaligned stores
def execute_sw( s, inst ):
    addr = trim_xlen( s.rf[inst.rs1] + inst.s_imm )
    if addr % 4 != 0: s.num_misaligned += 1
    s.mem.write( addr, 4, trim_32( s.rf[inst.rs2] ) )
    s.pc += 4

# record and count all executed loops
def execute_beq( s, inst ):
    if s.rf[inst.rs1] == s.rf[inst.rs2]:
        old_pc = s.pc
        s.pc = trim_xlen( s.pc + inst.sb_imm )
        if s.pc <= old_pc: s.loops[(s.pc, old_pc)] += 1
    else:
        s.pc += 4
```

# Pydgin in Our Research Group

---

- Statistics for software-defined regions
- Data-structure specialization experimentation
- Control- and memory-divergence for SIMD
- Basic Block Vector generation for SimPoint
- Analysis of JIT-enabled dynamic language interpreters





# Conclusions

---

Pydgin leverages the RPython translation toolchain into **high-performance**, DBT Instruction Set Simulator.

Pydgin provides a succinct architecture description language within Python to give users a **productive development, extension, and instrumentation** experience.

Current State: RV64IMAFD (RV64G) Bare-Metal on 64-bit host



Thank you to our sponsors for their support:  
NSF, DARPA, and donations from Intel Corporation and  
Synopsys, Inc.