DynamoRIO RISC-V Porting Progress

Yang Liu @ PLCT Lab
2023-12-15

Who am I

DynamoRIO RISC-V Porting Progress

Yang Liu
ksco

@plctlab

ptitSeb/**box64** Public

Box64 - Linux Userspace x86_64 Emulator with a twist, targeted at ARM64 Linux devices

C 2.7k 199

DynamoRIO/**dynamorio** Public

Dynamic Instrumentation Tool Platform

C 2.4k 546

Outline

DynamoRIO RISC-V Porting Progress

- Intro
- Usage
- How it works
- RV64 status

Outline

DynamoRIO RISC-V Porting Progress

- **Intro**
- Usage
- How it works
- RV64 status

Intro

DynamoRIO RISC-V Porting Progress

DynamoRIO is a runtime code manipulation system that supports code transformations on any part of a program, while it executes.

https://dynamorio.org/

Outline

DynamoRIO RISC-V Porting Progress

- Intro
- **Usage**
- How it works
- RV64 status

Usage

DynamoRIO RISC-V Porting Progress

```
USAGE: drrun [options] <app and args to run>
   or: drrun [options] -- <app and args to run>
   or: drrun [options] [DR options] -- <app and args to run>
   or: drrun [options] [DR options] -c <client> [client options] -- <app and args to run>
   or: drrun [options] [DR options] -t <tool> [tool options] -- <app and args to run>
   or: drrun [options] [DR options] -c32 <32-bit-client> [client options] -- -c64 <64-bit-
client> [client options] -- <app and args to run>
available tools include: drcachesim, drcov, drcpusim
```

Usage

DynamoRIO RISC-V Porting Progress

```
drrun -- echo hello
```

```
debian@revyos-pioneer:~/drinstall$ bin64/drrun -- echo hello
hello
```

Usage

DynamoRIO RISC-V Porting Progress

```
drrun -c libinscount.so -- echo hello
```

```
debian@revyos-pioneer:~/drinstall$ bin64/drrun -c samples/bin64/libinscount_test.so -- echo hello

Client inscount is running
hello
Instrumentation results: 243468 instructions executed
```

Usage

DynamoRIO RISC-V Porting Progress

```
drrun -t drcachesim -simulator_type reuse_distance -- echo hello
```

Reuse distance refers to the time interval between repeated accesses to the same data in a program.

Analyzing this pattern helps optimize cache performance.

Usage

DynamoRIO RISC-V Porting Progress

```
debian@revyos-pioneer:~/drinstall$ bin64/drrun -t drcachesim -simulator_type reuse_distance -- echo hello
hello
---- <application exited with code 0> ----
Reuse distance tool aggregated results:
Total accesses: 308824
Instruction accesses: 0
Data accesses: 308824
Unique accesses: 257373
Unique cache lines accessed: 4176
Distance limit: 0
Pruned addresses: 0
Pruned address hits: 0

Reuse distance mean: 17.98
Reuse distance median: 1
Reuse distance standard deviation: 93.85
(Pass -reuse_distance_histogram to see all the data.)

Reuse distance threshold = 100 cache lines
Top 10 frequently referenced cache lines
        cache line:     #references   #distant refs
      0x3fad40b2c0:        19746,            1
      0x3fcc49bec0:        12139,           27
      0x3fad4165c0:        10144,           94
      0x3facb77600:         9259,           23
      0x3fad40b300:         7444,            3
      0x3facb763c0:         5969,           26
      0x3fcc49bd00:         5964,           18
      0x3fad40eac0:         5808,            0
      0x3facb20740:         4230,           13
      0x3facb78400:         3980,           26
```

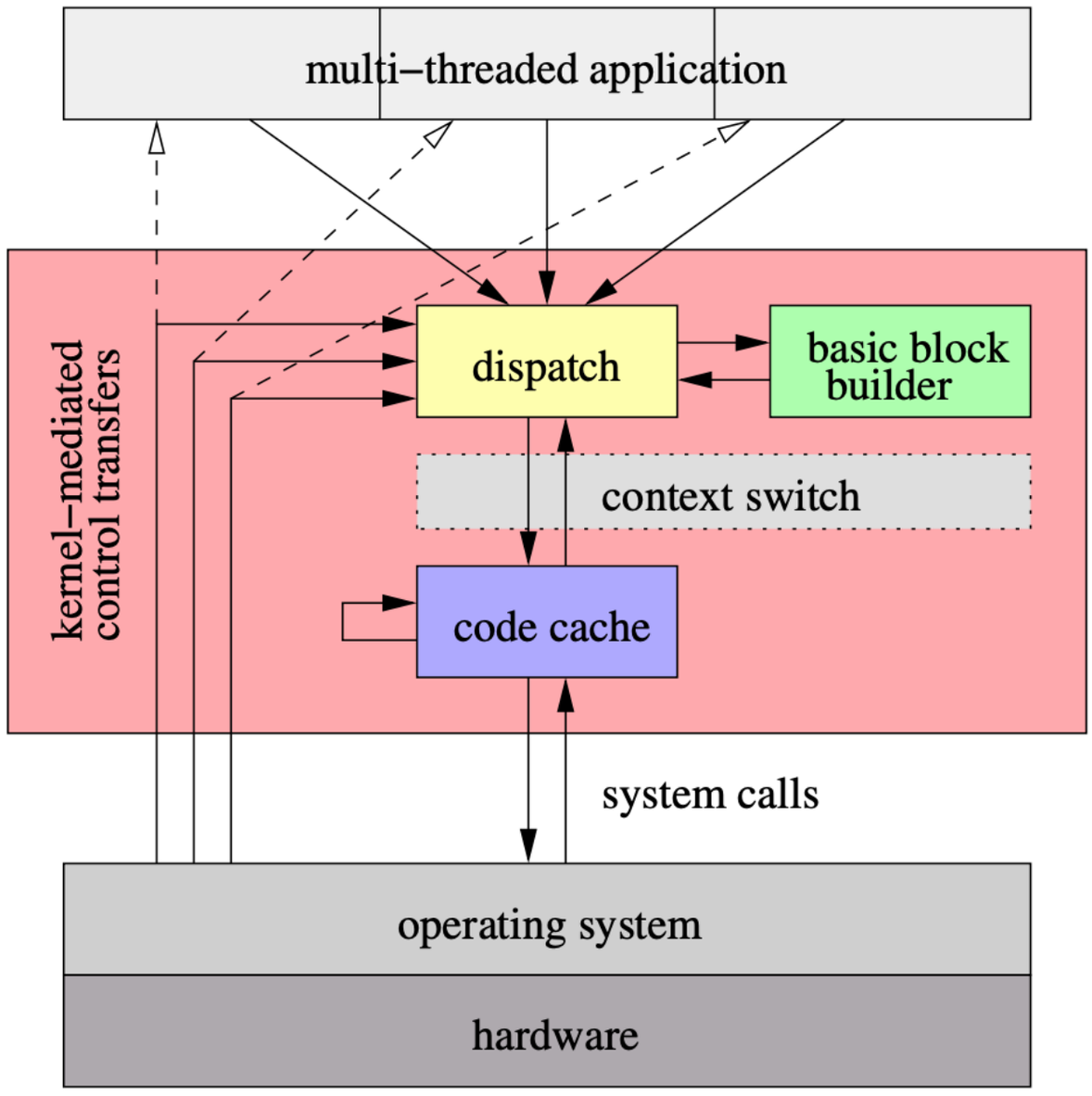
Outline

DynamoRIO RISC-V Porting Progress

- Intro
- Usage
- **How it works**
- RV64 status

How it works

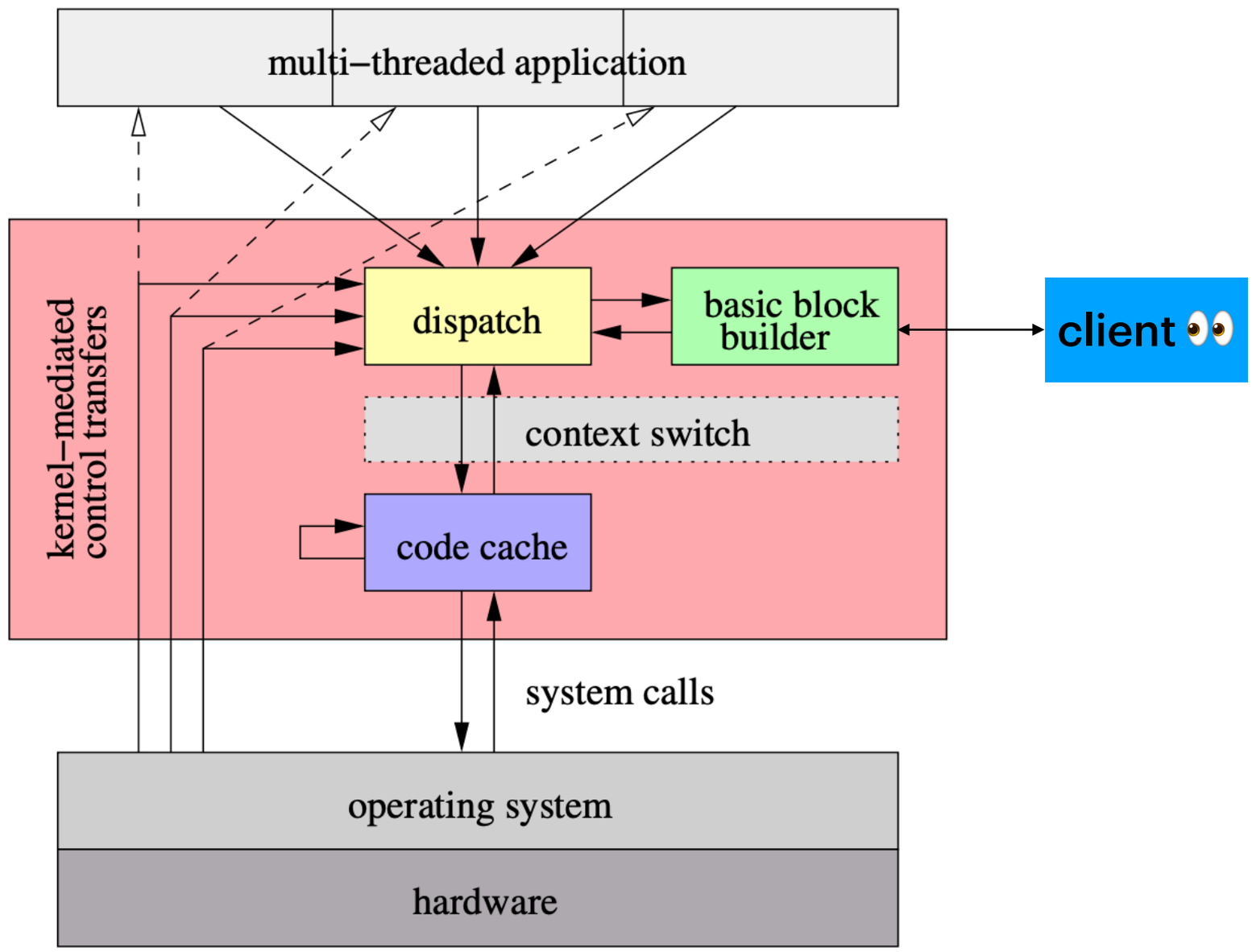
DynamoRIO RISC-V Porting Progress



Derek Bruening. Efficient, Transparent, and Comprehensive Runtime Code Manipulation. Ph.D. Thesis, MIT, September 2004.

How it works

DynamoRIO RISC-V Porting Progress



Derek Bruening. Efficient, Transparent, and Comprehensive Runtime Code Manipulation. Ph.D. Thesis, MIT, September 2004.

Outline

DynamoRIO RISC-V Porting Progress

- Intro
- Usage
- How it works
- **RV64 status**

RV64 status

DynamoRIO RISC-V Porting Progress

- x86
- x86-64
- ARM
- AArch64
- **RISCV64**

- Windows
- **Linux**
- Android
- macOS

RV64 status

DynamoRIO RISC-V Porting Progress

79 Total

Author Label Projects Milestones Reviews Assignee Sort

- i3544 RV64: Enable a few more sample tests ✕ — 1
  #6504 by ksco was merged 2 days ago • Approved
- i#3544 RV64: Enable sample.bbcount test ✕ — 1
  #6498 by ksco was merged last week • Approved
- i#3544 RV64: Enable sample.bbbuf test ✕ — 4
  #6493 by ksco was merged last week • Approved
- i#3544 RV64: Adds immediate display format via IR ✕ — 4
  #6489 by ksco was merged last week • Approved
- i#3544 RV64: Add support for XTheadCmo and XTheadSync extensions ✕ — 3
  #6477 by ksco was merged 2 weeks ago • Approved
- i#3544 RV64: Fix CI rsync issue ✕ — 2
  #6476 by derekbruening was merged 2 weeks ago • Approved
- i#3544 RV64: Refine fault translation and signal handling ✕ — 19
  #6461 by ksco was merged 2 weeks ago • Approved
- i#3544 RV64: Enable more tests on CI ✓ — 6
  #6447 by ksco was merged on Nov 14 • Approved
- i#3544 RV64: Fill in the missing pieces to run actual programs ✓ — 13
  #6437 by ksco was merged on Nov 11 • Approved
- i#3544 RV64: Fix a copy-paste mistake in execute_handler_from_dispatch ✓ — 1

RV64 status

DynamoRIO RISC-V Porting Progress

- **5** contributors by far on RISCV64
- **79** PRs merged in total, 48 by me
- **several** upstream reviewers
- we're hiring interns: plctlab/weloveinterns BJ107

RV64 status

DynamoRIO RISC-V Porting Progress

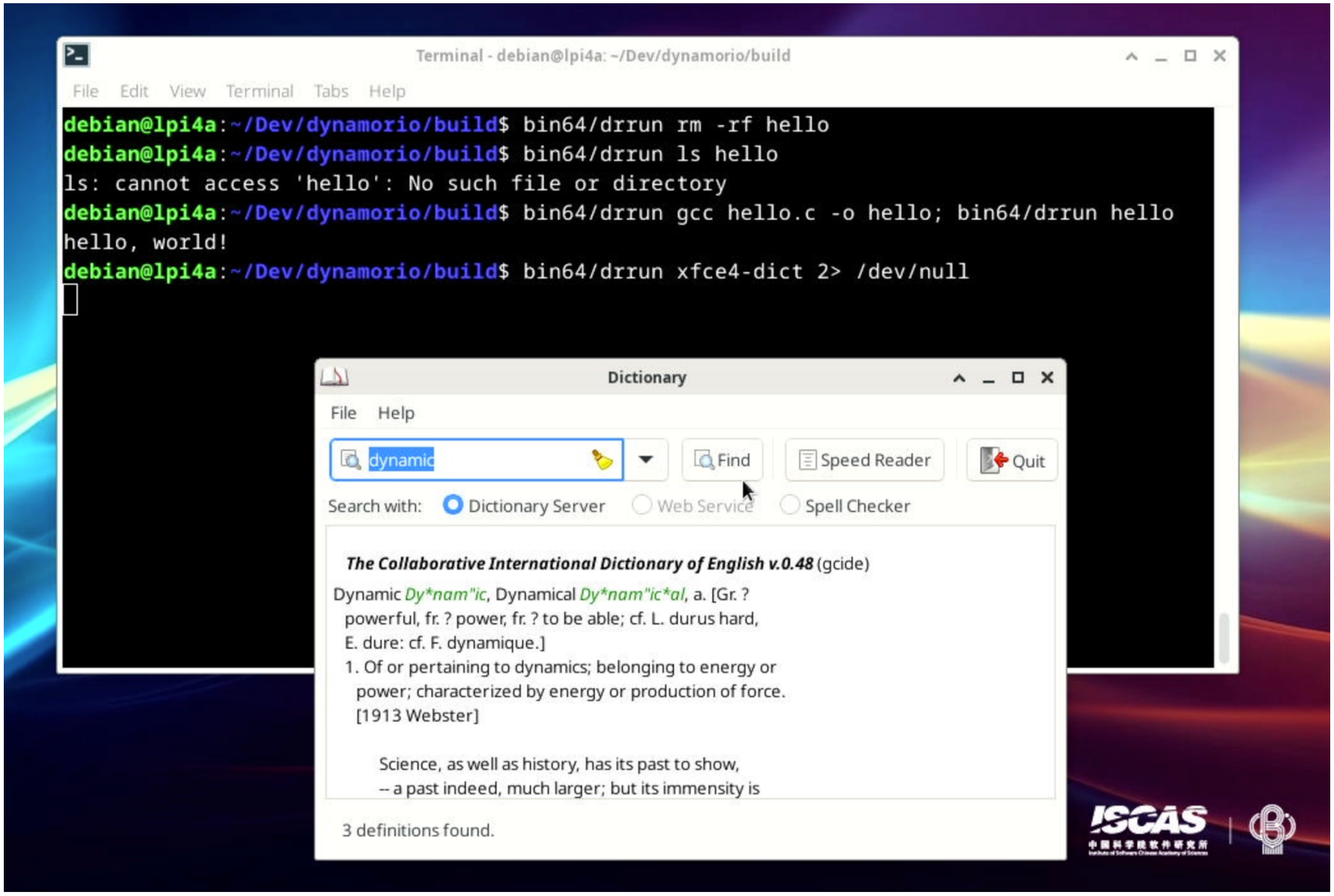
```
debian@revyos-pioneer:~/drinstall$ cat hello.c
#include <stdio.h>

int main() {
    printf("hello\n");
    return 0;
}
debian@revyos-pioneer:~/drinstall$ bin64/drrun -c samples/bin64/libinscount_test.so -- gcc hello.c
<Starting application /usr/bin/riscv64-linux-gnu-gcc-13 (3466)>
Client inscount is running
<-- execve /usr/libexec/gcc/riscv64-linux-gnu/13/cc1 -->
<-- execve /usr/local/bin/as -->
<-- execve /usr/bin/as -->
<-- execve /usr/libexec/gcc/riscv64-linux-gnu/13/collect2 -->
<Stopping application /usr/bin/riscv64-linux-gnu-gcc-13 (3466)>
Instrumentation results: 2041855 instructions executed

debian@revyos-pioneer:~/drinstall$ ./a.out
hello
debian@revyos-pioneer:~/drinstall$
```

RV64 status

DynamoRIO RISC-V Porting Progress



RV64 status

DynamoRIO RISC-V Porting Progress

(known) RV64 unimplemented features

- clean call optimisation
- trace building
- thread-private fragment cache
- most of the tools are not ported
- some samples are not ported
- some tests are not ported
- Android
- Dr.Memory
- encoder/decoder: more extensions to be done
- ...