LLVM Advisor

A Unified Visualization Tool for Compiler Artifacts

Miguel Cárdenas | Google Summer of Code 2025

Mentored by Jose M. Monsalve Diaz[1], Kevin Sala[2] and Johannes Doerfert[2]

[1] Advanced Micro Devices
[2] Lawrence Livermore National Laboratory

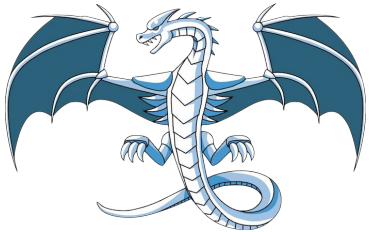
2025 LLVM Developers' Meeting
October 28, 2025

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-PRES-2012797.

Context

Context

- **LLVM** reports **valuable compilation data**
  - Optimization remarks
  - Profiling data
  - Timing data



Context

- **LLVM** reports **valuable compilation data**
  - Optimization remarks
  - Profiling data
  - Timing data

- However…
  - Multiple compilation **flags**
  - Different **output data formats**
  - **Finding remarks** in specific regions

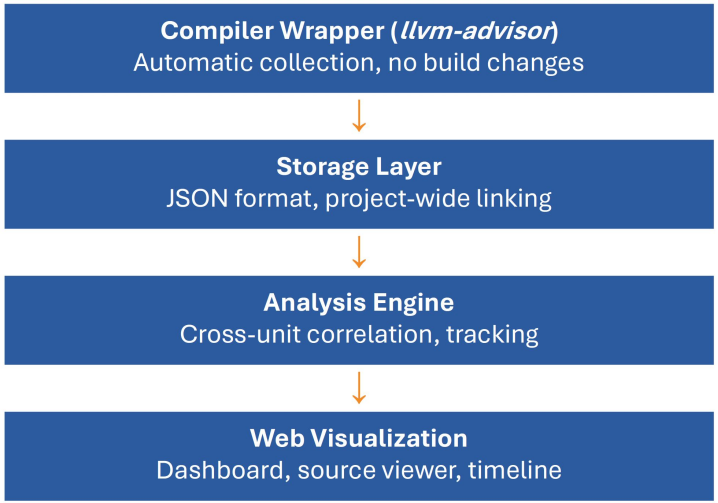
```
remark: in artificial function
      '__omp_offloading_803_29744d_main_l9_omp_outlined_omp_outlined',
      alloca ('%15') without debug info with static size of 4 bytes
      [-Rpass=kernel-info]
test.c:9:5: remark: in artificial function
      '__omp_offloading_803_29744d_main_l9_omp_outlined_omp_outlined',
      direct call, callee is '@llvm.lifetime.start.p0' [-Rpass=kernel-info]
    9 |     #pragma omp target teams distribute parallel for map(to: a, b) m
...
      |     ^
test.c:9:5: remark: in artificial function
      '__omp_offloading_803_29744d_main_l9_omp_outlined_omp_outlined',
      direct call, callee is '@llvm.lifetime.start.p0' [-Rpass=kernel-info]
test.c:9:5: remark: in artificial function
      '__omp_offloading_803_29744d_main_l9_omp_outlined_omp_outlined',
      direct call, callee is '@llvm.lifetime.start.p0' [-Rpass=kernel-info]
test.c:9:5: remark: in artificial function
      '__omp_offloading_803_29744d_main_l9_omp_outlined_omp_outlined',
      direct call, callee is '@llvm.lifetime.start.p0' [-Rpass=kernel-info]
test.c:9:5: remark: in artificial function
      '__omp_offloading_803_29744d_main_l9_omp_outlined_omp_outlined',
      'load' instruction ('%25') accesses memory in flat address space
      [-Rpass=kernel-info]
test.c:9:5: remark: in artificial function
      '__omp_offloading_803_29744d_main_l9_omp_outlined_omp_outlined',
      direct call, callee is '@__kmpc_for_static_init_4'
      [-Rpass=kernel-info]
test.c:11:16: remark: in artificial function
      '__omp_offloading_803_29744d_main_l9_omp_outlined_omp_outlined',
```

```
 "/home/cc/Files/llvm/bin/clang-offload-packager" "-o" "/tmp/test-f38eb3.out
" "--image=file=/tmp/test-sm_75-7a20ba.o,triple=nvptx64-nvidia-cuda,arch=sm_
75,kind=openmp,feature=+ptx85"
 "/home/cc/Files/llvm/bin/clang-21" "-cc1" "-triple" "x86_64-unknown-linux-g
nu" "-O2" "-emit-obj" "-dumpdir" "a-" "-disable-free" "-clear-ast-before-bac
kend" "-disable-llvm-verifier" "-discard-value-names" "-main-file-name" "tes
t.c" "-mrelocation-model" "pic" "-pic-level" "2" "-pic-is-pie" "-mframe-poin
ter=none" "-fmath-errno" "-ffp-contract=on" "-fno-rounding-math" "-mconstruc
tor-aliases" "-funwind-tables=2" "-target-cpu" "x86-64" "-tune-cpu" "generic
" "-debug-info-kind=constructor" "-dwarf-version=5" "-debugger-tuning=gdb" "
-fdebug-compilation-dir=/home/cc/testing" "-fcoverage-compilation-dir=/home/
cc/testing" "-resource-dir" "/home/cc/Files/llvm/lib/clang/21" "-ferror-limi
t" "19" "-fmessage-length=76" "-fopenmp" "-fgnuc-version=4.2.1" "-fskip-odr-
check-in-gmf" "-fcolor-diagnostics" "-vectorize-loops" "-vectorize-slp" "-fe
mbed-offload-object=/tmp/test-f38eb3.out" "--offload-targets=nvptx64-nvidia-
cuda" "-faddrsig" "-D__GCC_HAVE_DWARF2_CFI_ASM=1" "-o" "/tmp/test-835b01.o"
"-x" "ir" "/tmp/test-8a78b4.bc"
```

Solution: LLVM Advisor

LLVM Advisor

- **LLVM Advisor** provides a **unified infrastructure**
  - **Gather** and **organize** compilation data
  - **Analyze** and correlate data
  - **Visualize** on a modern web interface



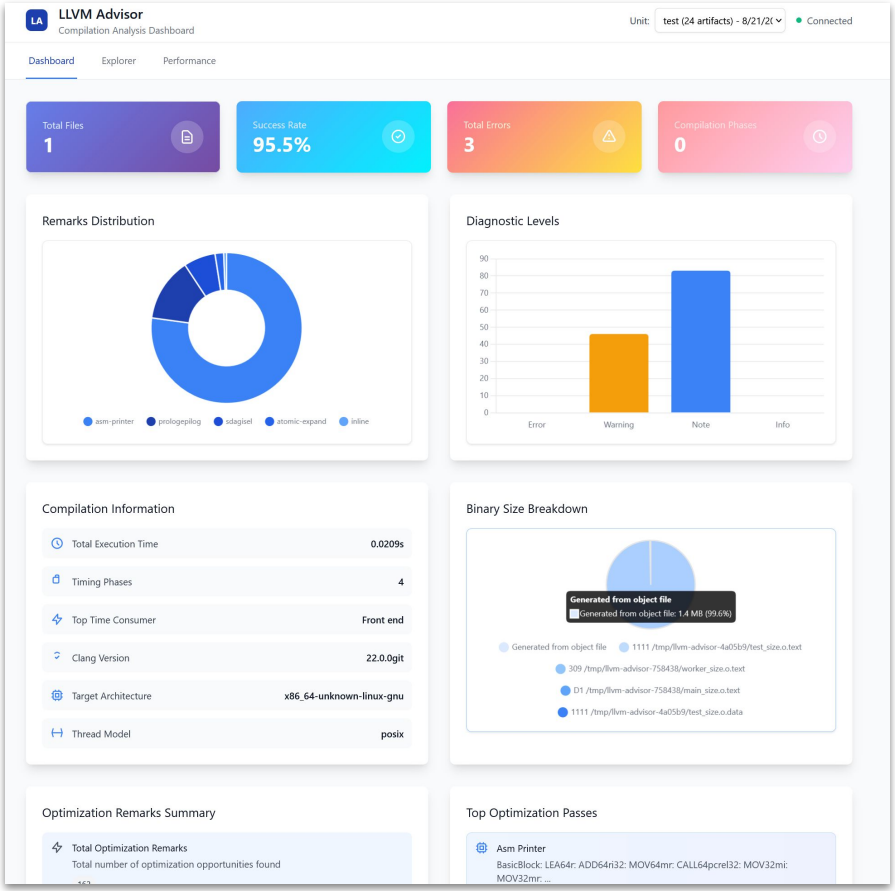
LLVM Advisor

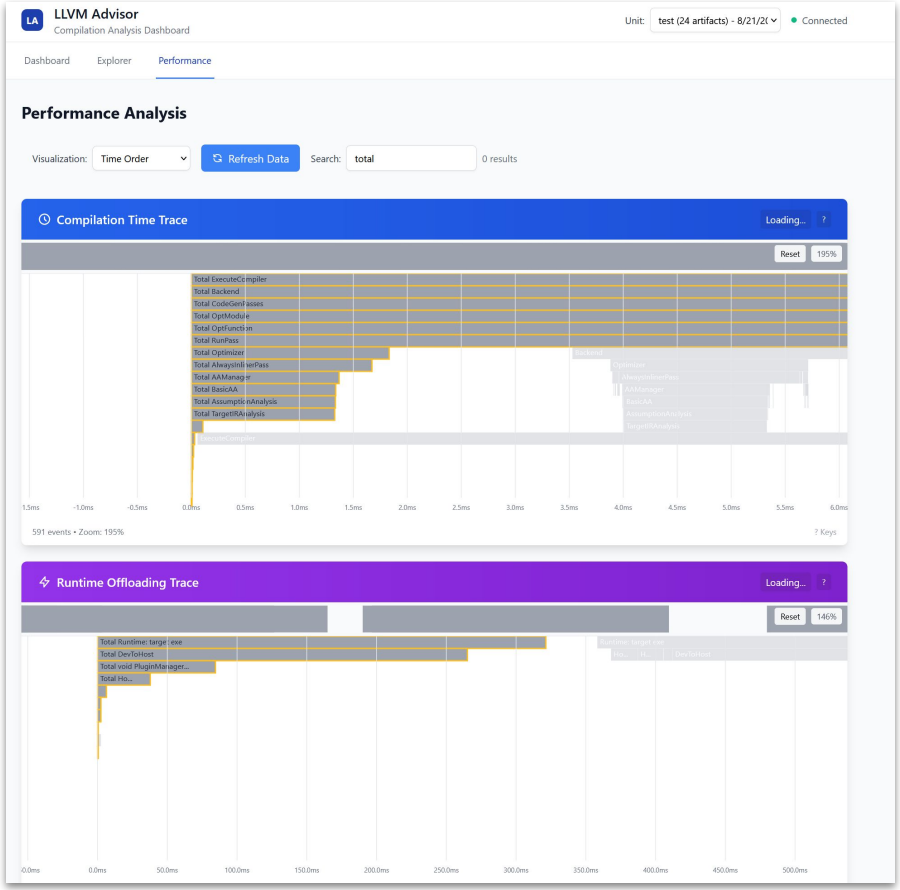
- Compile with data collection

```
$ llvm-advisor clang -O2 -g main.c -o main
```

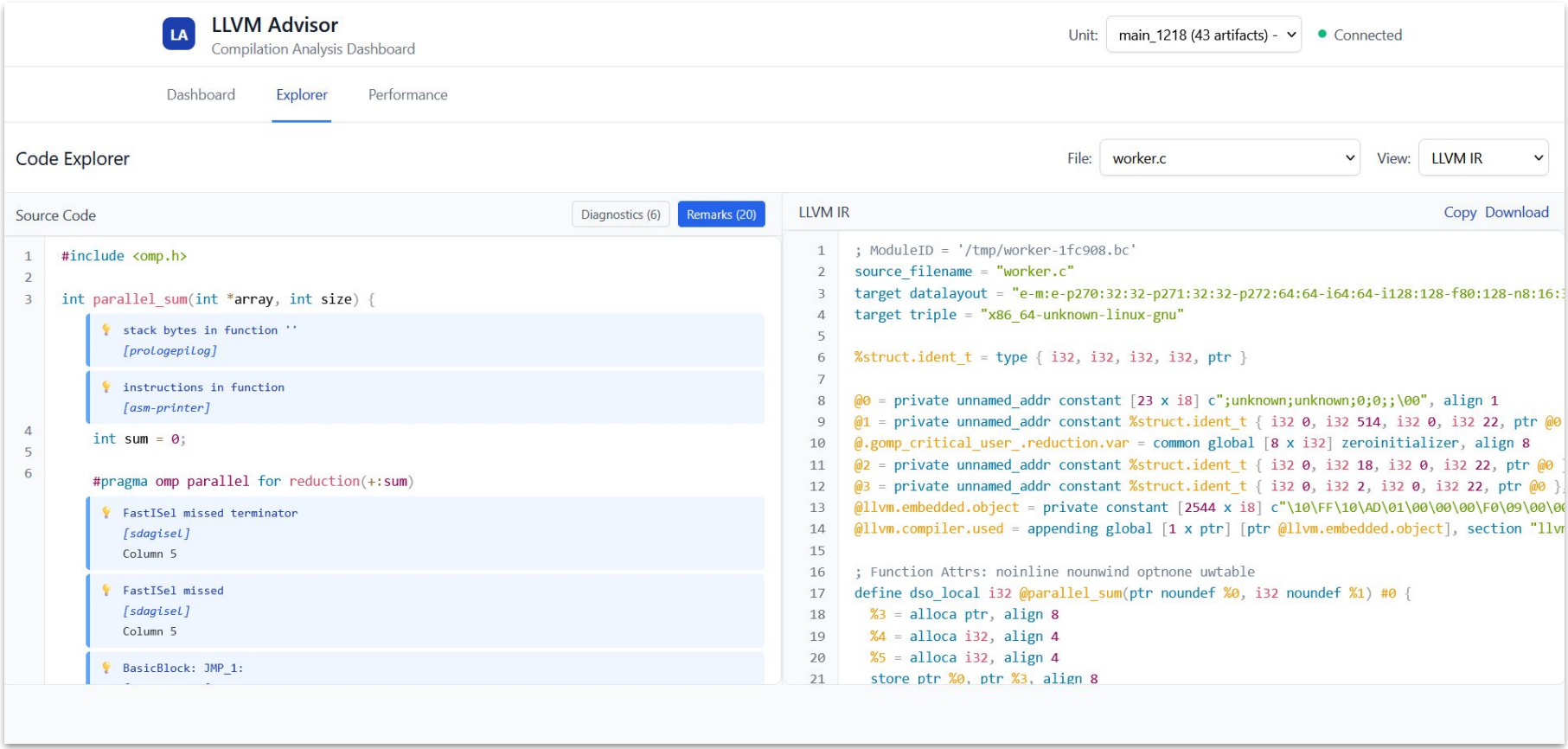
```
$ tree -a .llvm-advisor
.llvm-advisor/
└── main/
    └── main_20250825_143022/  # Timestamped compilation session
        ├── ir/main.ll         # LLVM IR output
        ├── assembly/main.s    # Assembly output
        ├── ast/main.ast       # AST dump
        ├── diagnostics/       # Compiler warnings/errors
        └── ...                # Additional analysis data
```

LLVM Advisor

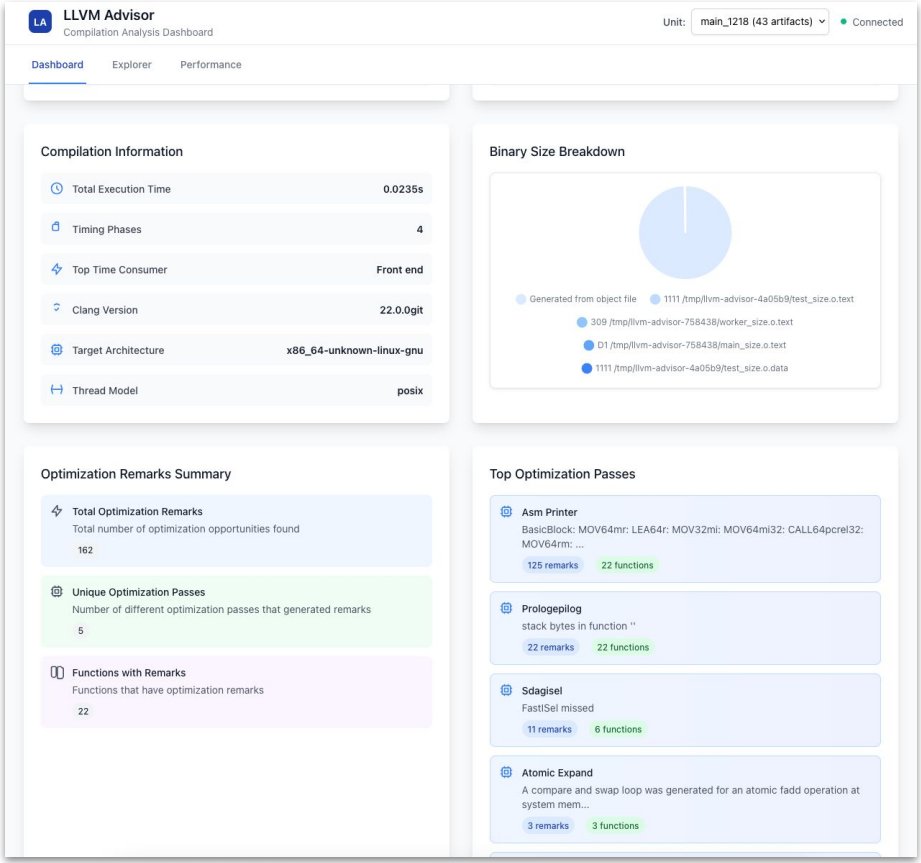




LLVM Advisor



LLVM Advisor



- Who can benefit?
  - **Advanced developers**
  - **Newcomers** in LLVM
  - **Educators**

Questions?

Visit the poster session tomorrow!

Miguel Cárdenas
Student at University of Medellin
miguelecsx@gmail.com
@miguelcsx | LLVM, GitHub, LinkedIn

Try LLVM Advisor

https://llvm-advisor.onrender.com/

LLVM PR #147451
https://github.com/llvm/llvm-project/pull/147451