Triton Update

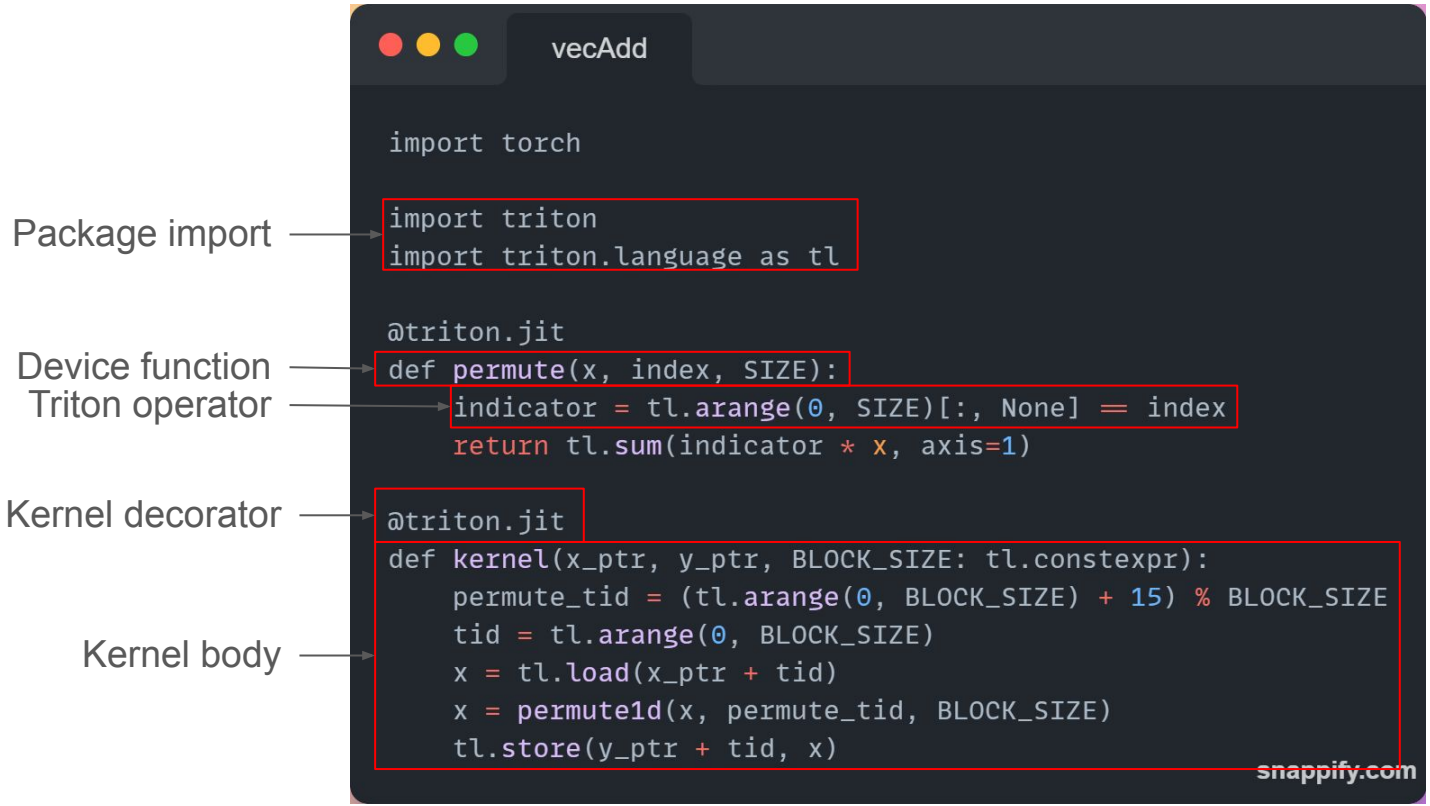
Keren Zhou
kzhou6@gmu.edu

Triton Review

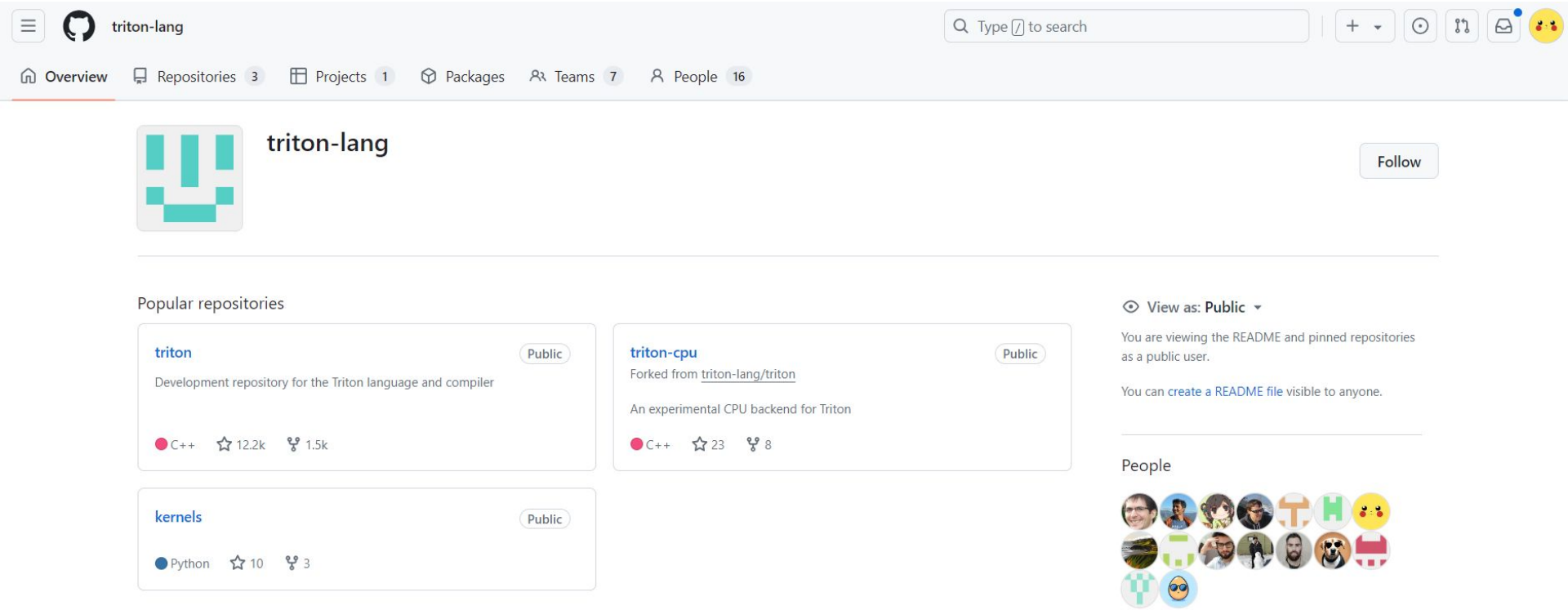
What is Triton

- A Python-like language
- A JIT compiler
- A PyTorch backend
- A set of MLIR dialects
- A community
- An organization

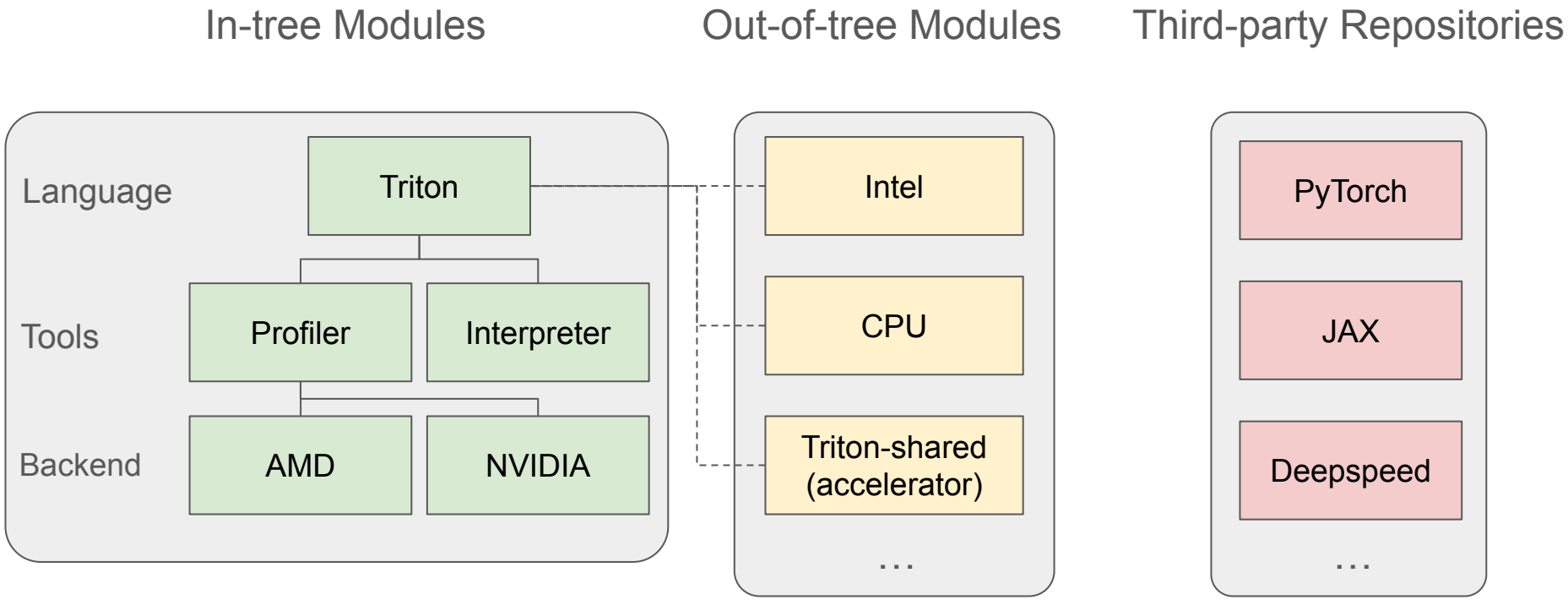
A Triton Program (Permutation)



The Triton-Lang Organization



Triton Community



Beginner’s Resources

- Triton Index
  - cuda-mode/triton-index: Cataloging released Triton kernels. (github.com)
- Awesome Triton Kernels
  - zinccat/Awesome-Triton-Kernels: Collection of kernels written in Triton language (github.com)
- Unsloth
  - unslothai/unsloth: Finetune Llama 3, Mistral & Gemma LLMs 2-5x faster with 80% less memory (github.com)
- Triton Puzzles
  - srush/Triton-Puzzles: Puzzles for learning Triton (github.com)
- Torchao
  - pytorch/ao: Native PyTorch library for quantization and sparsity (github.com)
- Attorch
  - BobMcDear/attorch: A subset of PyTorch's neural network modules, written in Python using OpenAI's Triton. (github.com)

Guide for Developers

- Read the Triton source code!
- Read the MLIR source code!
- I found a handful of Triton backend analysis articles on zhihu.com
  - But triton core developers may not have time to write any of these
  - We prefer to leaving comments to save time
  - Discussion
    - [QST] Triton MLIR · Issue #3 · srush/Triton-Puzzles (github.com)

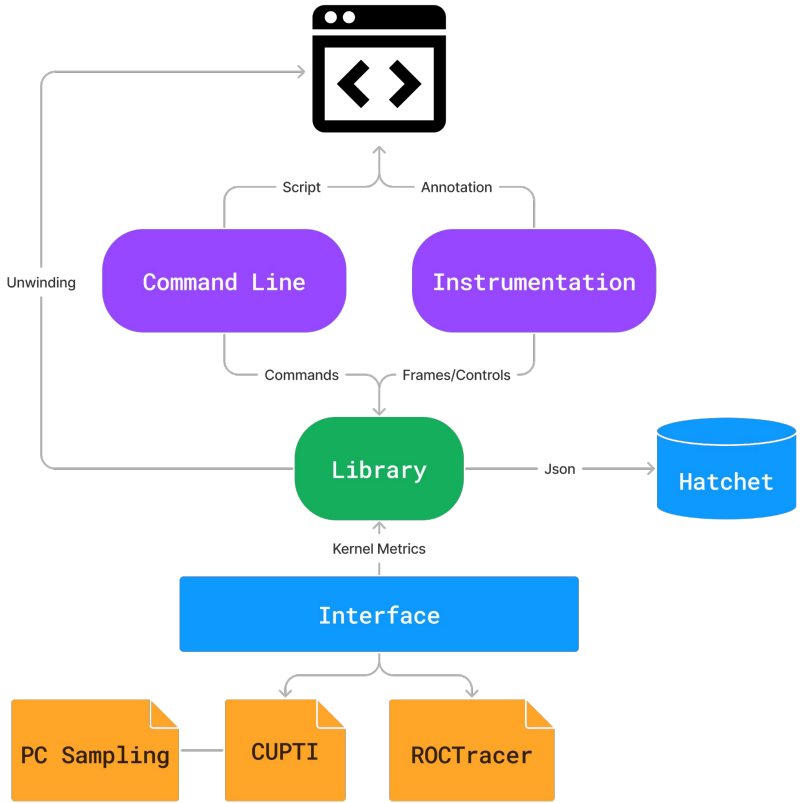
Proton

A Pro**filer for Trit**on

Proton

- Provide a quick, intuitive, and simple way to check kernel performance
  - Open source
  - Multiple vendor GPUs
  - Flexible metrics collection
    - Hardware metrics
    - Software metrics
  - Call path profiling

Design


Script
Annotation
Unwinding
Command Line
Instrumentation
Commands
Frames/Controls
Library
Json
Hatchet
Kernel Metrics
Interface
PC Sampling
CUPTI
ROCTracer

Call Path Profiling

- Profile kernel running time

```
55.193 ROOT
├─ 31.212 /home/kzhou6/gh200/triton/third_party/proton/tutorials/dynamic_net.py:<module>@98
|  └─ 31.212 /home/kzhou6/gh200/triton/python/triton/profiler/profile.py:wrapper@151
|     ├─ 0.002 /home/kzhou6/gh200/triton/third_party/proton/tutorials/dynamic_net.py:run@51
|     |  └─ 0.002 _ZN50_GLOBAL__N__c922cf59_17_RangeFactories_cu_38772b0829elementwise_kernel_with_indexIi
_clEvENKUlvE0_clEvEUllE_EEvT_T0_PN15function_traitsISD_E11result_typeE
|     ├─ 0.003 /home/kzhou6/gh200/triton/third_party/proton/tutorials/dynamic_net.py:run@52
|     |  └─ 0.003 _ZN2at6native29vectorized_elementwise_kernelILi4EZZZNS0_15sin_kernel_cudaERNS_18TensorIt
_T1_
|     ├─ 19.610 /home/kzhou6/gh200/triton/third_party/proton/tutorials/dynamic_net.py:run@66
|     |  └─ 19.610 /home/kzhou6/gh200/pytorch/torch/nn/modules/module.py:_wrapped_call_impl@1532
|     |     └─ 19.610 /home/kzhou6/gh200/pytorch/torch/nn/modules/module.py:_call_impl@1541
|     |        ├─ 13.931 /home/kzhou6/gh200/triton/third_party/proton/tutorials/dynamic_net.py:forward@36
|     |        |  ├─ 2.939 /home/kzhou6/gh200/pytorch/torch/_tensor.py:wrapped@40
|     |        |  |  ├─ 1.460 _ZN2at6native29vectorized_elementwise_kernelILi4EZNS0_53_GLOBAL__N__2ced54f0
18TensorIteratorBaseET0_EUlfE0_NS_6detail5ArrayIPcLi2EEEEEviS6_T1_
|     |        |  |  └─ 1.479 _ZN2at6native29vectorized_elementwise_kernelILi4EZNS0_53_GLOBAL__N__2ced54f0
18TensorIteratorBaseET0_EUlfE_NS_6detail5ArrayIPcLi2EEEEEviS6_T1_
|     |        |  ├─ 6.022 _ZN2at6native18elementwise_kernelILi128ELi2EZNS0_22gpu_kernel_impl_nocastINS0_1
eratorBaseERKT_EUliE_EEviT1_
|     |        |  |  ├─ 2.025 _ZN2at6native18elementwise_kernelILi128ELi2EZNS0_22gpu_kernel_impl_nocastINS0_1
|     |        |  |  └─ 2.945 _ZN2at6native29vectorized_elementwise_kernelILi4ENS0_15CUDAFunctor_addIfEENS_6d
```

Python Context

```
54.763 ROOT
├─ 25.004 backward
|  ├─ 14.366 _ZN2at6native13reduce_kernelILi512ELi1ENS0_
|  ├─ 2.007 _ZN2at6native18elementwise_kernelILi128ELi2E
vEUlfffE_EEvRNS_18TensorIteratorBaseERKT_EUliE_EEviT1_
|  ├─ 2.461 _ZN2at6native29vectorized_elementwise_kernel
|  ├─ 5.725 _ZN2at6native29vectorized_elementwise_kernel
|  └─ 0.446 _ZN2at6native29vectorized_elementwise_kernel
├─ 19.399 forward
|  ├─ 7.961 _ZN2at6native18elementwise_kernelILi128ELi2E
EUliE_EEviT1_
|  ├─ 2.018 _ZN2at6native18elementwise_kernelILi128ELi2E
|  ├─ 4.415 _ZN2at6native29vectorized_elementwise_kernel
|  ├─ 1.455 _ZN2at6native29vectorized_elementwise_kernel
seET0_EUlfE0_NS_6detail5ArrayIPcLi2EEEEEviS6_T1_
|  ├─ 2.073 _ZN2at6native29vectorized_elementwise_kernel
seET0_EUlfE2_NS_6detail5ArrayIPcLi2EEEEEviS6_T1_
|  └─ 1.477 _ZN2at6native29vectorized_elementwise_kernel
seET0_EUlfE_NS_6detail5ArrayIPcLi2EEEEEviS6_T1_
├─ 0.004 init
|  ├─ 0.003 _ZN2at6native29vectorized_elementwise_kernel
|  └─ 0.001 _ZN50_GLOBAL__N__c922cf59_17_RangeFactories_
NKUlvE0_clEvEUllE_EEvT_T0_PN15function_traitsISD_E11resu
├─ 4.412 loss
|  ├─ 2.949 _ZN2at6native13reduce_kernelILi512ELi1ENS0_8
|  └─ 1.462 _ZN2at6native29vectorized_elementwise_kernel
```

Shadow Context

User Interface

- Lightweight source code instrumentation
  - Profile start/stop/finalize
  - Scopes
  - Hooks
- Command line
  - `python -m proton main.py`
  - `proton main.py`

Profile Start/Stop/Finalize

- Profile only interesting regions
  - `proton.start(profile_name: str) -> session_id: int`
  - `proton.finalize()`
- Skip some regions, but accumulate to the same profile
  - `session_id = proton.start(...)`
  - `proton.deactive(session_id)`
  - `… # region skipped`
  - `proton.activate(session_id)`

Scopes

- Only collect the *Master Thread* scope
  - In PyTorch, the thread that train and test models

```
with proton.scope("test0"):
    with proton.scope("test1"):
        foo[1,](x, y)
with proton.scope("test2"):
    foo[1,](x, y)
```

```
4544.000 ROOT
├─ 1472.000 _ZN2at6native29vectorized_elementwise_kernelILi
├─ 1664.000 test0
|  └─ 1664.000 test1
|     └─ 1664.000 foo
└─ 1408.000 test2
   └─ 1408.000 foo

Legend (Metric: Time (ns) (inc) Min: 1408.00 Max: 4544.00)
4230.40 - 4544.00
3603.20 - 4230.40
2976.00 - 3603.20
2348.80 - 2976.00
1721.60 - 2348.80
1408.00 - 1721.60
```

Metrics

- Asynchronous metrics
  - Come from profilers
- Synchronous metrics
  - Come from users
    - Theoretical flops, bytes
    - Loss
    - Counts
  - `Dict[str, Union[int, float]]`

```
with proton.scope("test0", {"foo_metric": 1.0}):
    foo[1,](x, y)
```

"test0" scope ends with multiple metrics.
Two metrics can be displayed the same time.

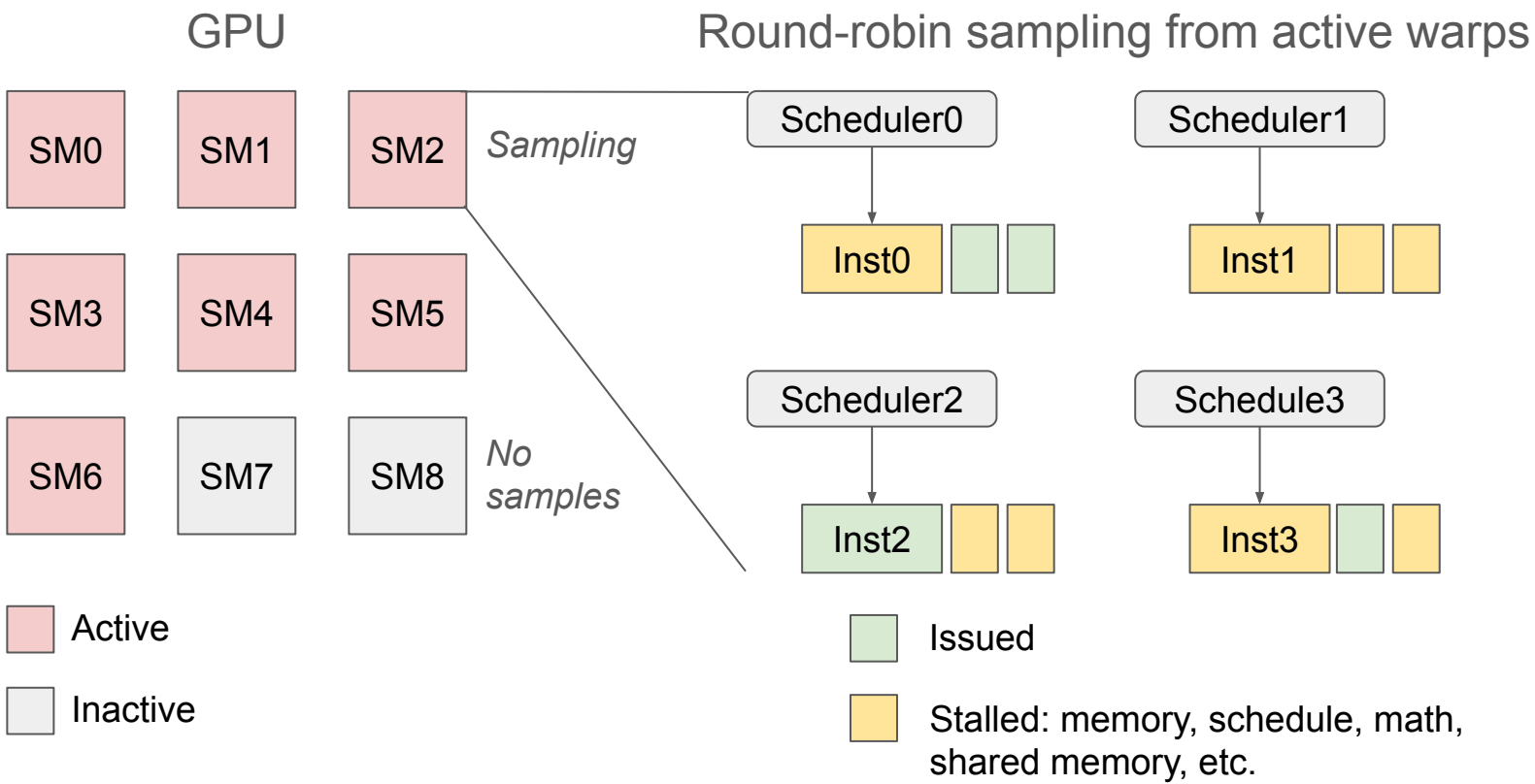
```
(pytorch) kzhou6@gracehopper:~$ proton-viewer -l proton.hatchet
<IPython.core.display.Javascript object>
Warning: Roundtrip module could not be loaded. Requires jupyter
Available metrics:
- count
- time
- foo_metric
(pytorch) kzhou6@gracehopper:~$ proton-viewer -m time,foo_metric
<IPython.core.display.Javascript object>
Warning: Roundtrip module could not be loaded. Requires jupyter
3104.000 1.000 ROOT
├─ 1440.000 nan _ZN2at6native29vectorized_elementwise_kernelILi4
└─ 1664.000 1.000 test0
   └─ 1664.000 nan foo

Legend (Metric: Time (ns) (inc) Min: 1440.00 Max: 3104.00)
2937.60 - 3104.00
2604.80 - 2937.60
2272.00 - 2604.80
1939.20 - 2272.00
1606.40 - 1939.20
1440.00 - 1606.40
```

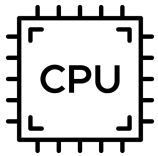
Triton Hooks

- Rename the triton function with a custom name
  - Append launch configurations
  - Append runtime dynamic
  - Append constants
  - e.g., `foo_<num_warps:4>_<fast_math:4>_<branch_0:1>`
- Supply custom metrics based on kernel arguments
  - flops{8, 16, 32, 64}
  - bytes

Instruction Sampling on NVIDIA GPUs

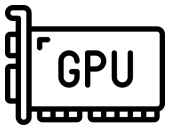


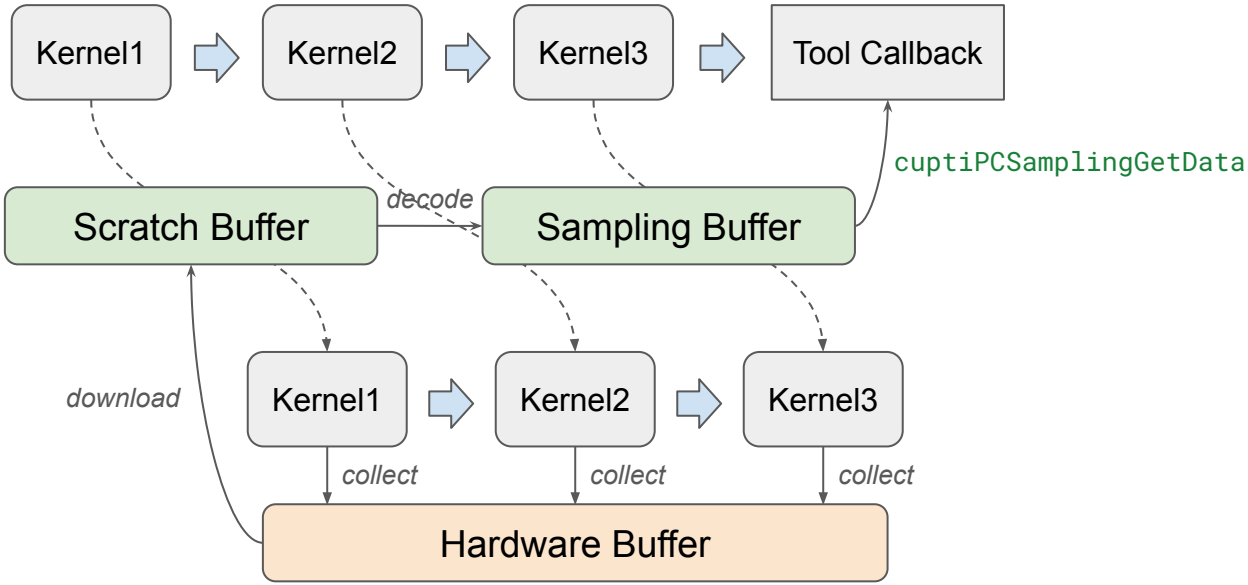
CUPTI Internals



Application Thread:

CUPTI Thread:





Proton's View

- Total samples and stalled samples

```
├─ 8483.000 5349.000 matmul_<grid:18x1x1>_<cluster:1x1x1>_<warps:8>_<shared:147456>_<stages:3>
|  ├─ 1080.000 728.000 /home/kzhou6/gh200/triton/third_party/proton/tutorials/matmul.py:matmul_kernel@107
|  ├─ 6793.000 4011.000 /home/kzhou6/gh200/triton/third_party/proton/tutorials/matmul.py:matmul_kernel@111
|  ├─ 488.000 488.000 /home/kzhou6/gh200/triton/third_party/proton/tutorials/matmul.py:matmul_kernel@116
|  └─ 122.000 122.000 /home/kzhou6/gh200/triton/third_party/proton/tutorials/matmul.py:matmul_kernel@96
```

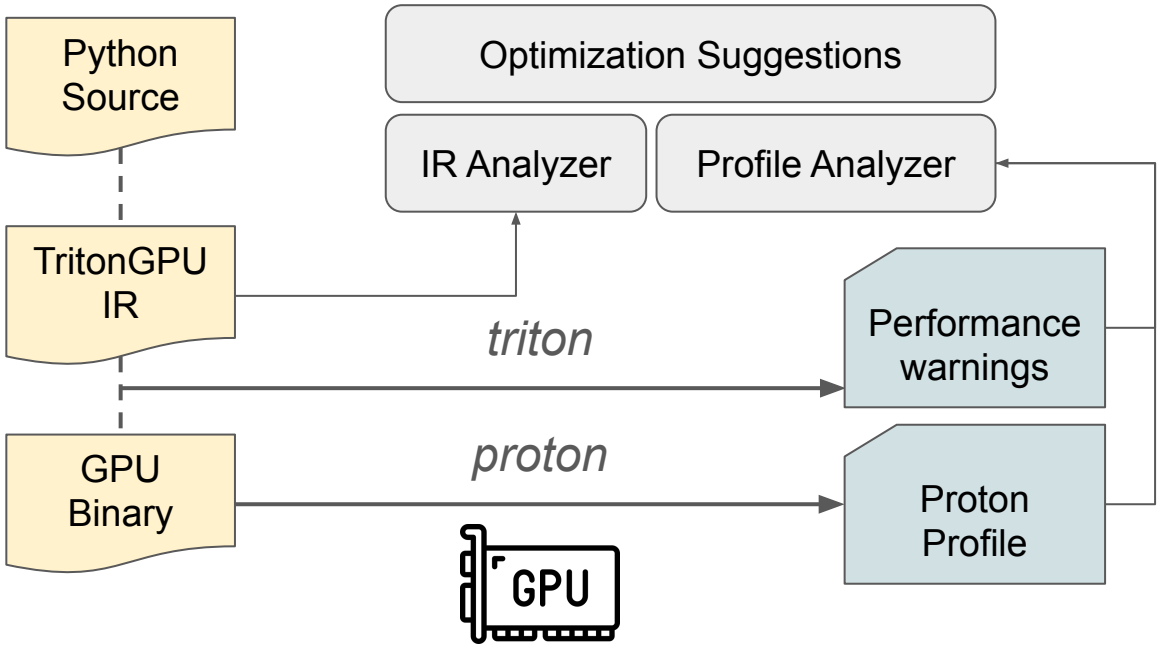
```
106     accumulator = tl.zeros((BLOCK_SIZE_M, BLOCK_SIZE_N), dtype=tl.float32)
107     for k in range(0, tl.cdiv(K, BLOCK_SIZE_K)):
108         # Load the next block of A and B, generate a mask by checking the K dimension.
109         # If it is out of bounds, set it to 0.
110         a = tl.load(a_ptrs, mask=offs_k[None, :] < K - k * BLOCK_SIZE_K, other=0.0)
111         b = tl.load(b_ptrs, mask=offs_k[:, None] < K - k * BLOCK_SIZE_K, other=0.0)
112         # We accumulate along the K dimension.
113         accumulator += tl.dot(a, b)
114         # Advance the ptrs to the next K block.
115         a_ptrs += BLOCK_SIZE_K * stride_ak
116         b_ptrs += BLOCK_SIZE_K * stride_bk
```

Overhead

- NCU overhead *>1000x*
  - *4s -> 66 mins*
  - `time ncu --section SourceCounters python ./dynamic_net.py`
- Proton overhead *~20x*
  - Could be reduced to less than *5x*
    - Many optimizations haven’t been applied

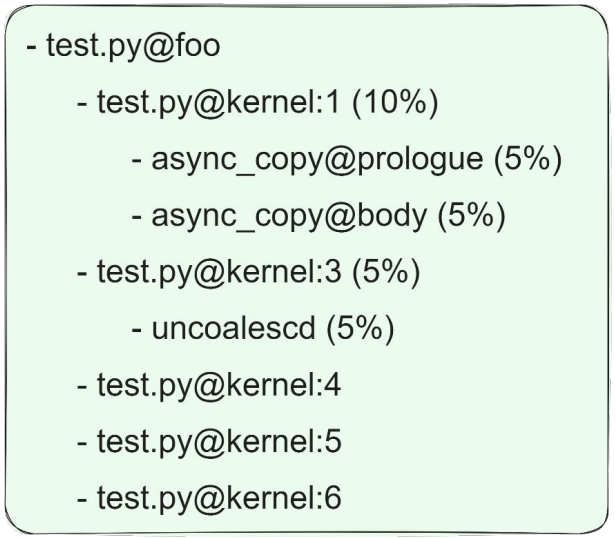
Proton-Analyzer

- Design for Torchinductor and Triton

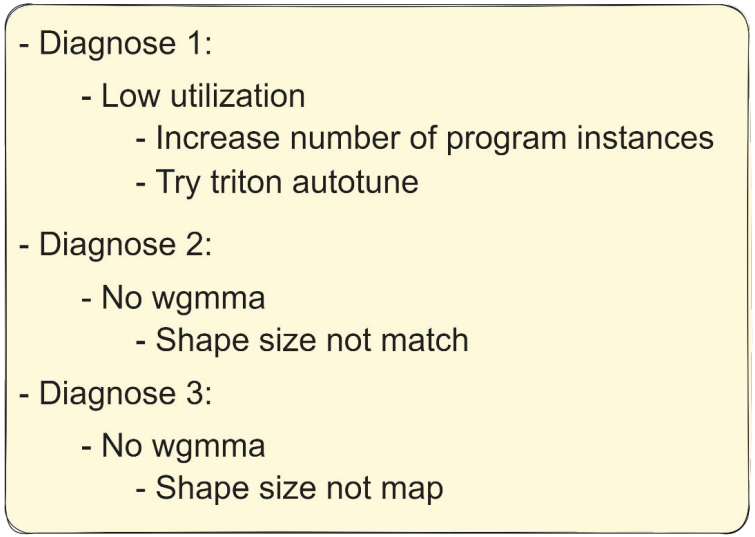


Potential Views

- Interactive view
  - https://godbolt.org
- Terminal view



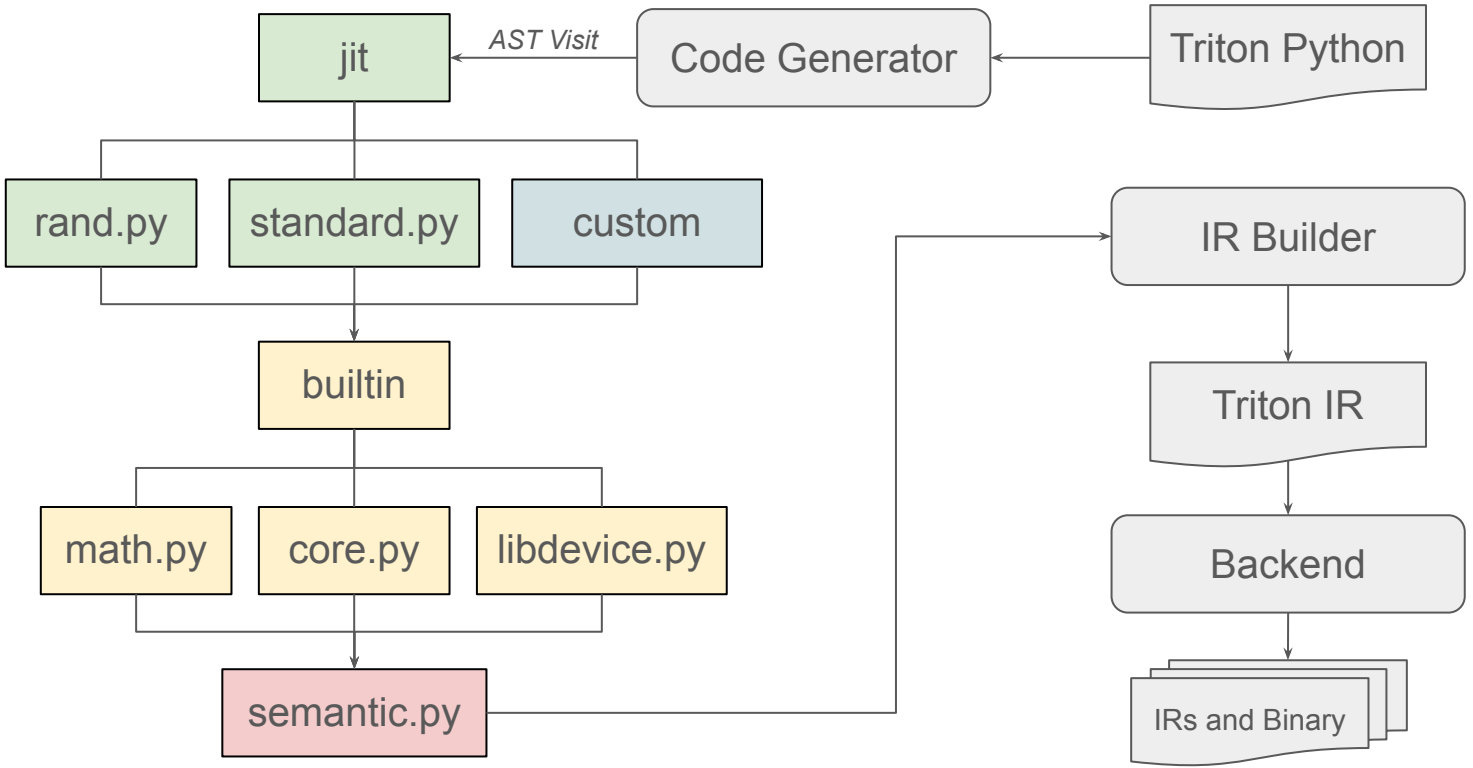
Profile Report



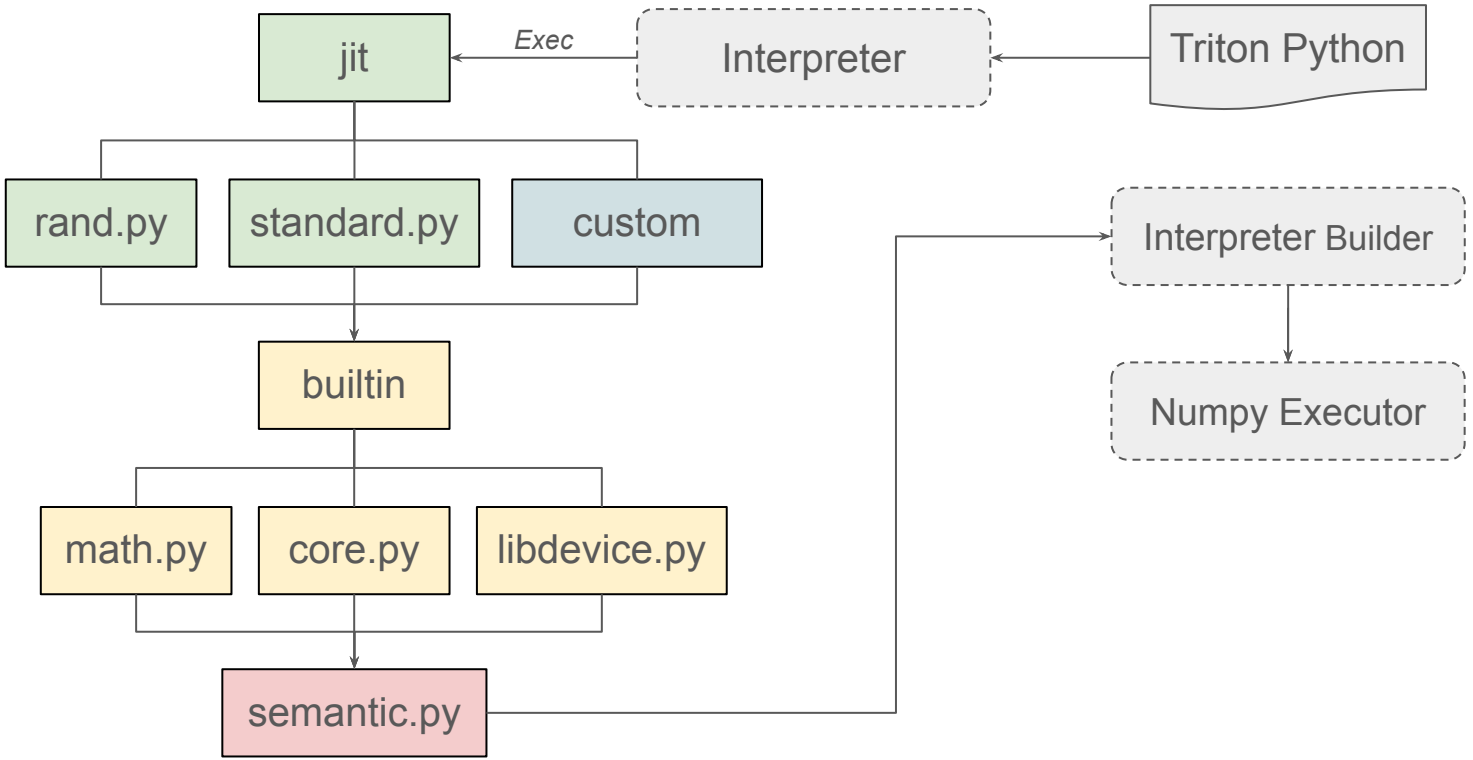
Diagnostic Report

Interpreter

Revisit the Frontend



Interpreter



Usage

- Enable the interpreter mode
  - `TRITON_INTERPRET=1 <your command>`
- Debug with `pdb`
  - `TRITON_INTERPRET=1 pdb test.py`
  - `b test.py:<line number>`
  - `r`
- Highlights
  - You can set `device='cpu'` to execute code with the interpreter
  - You can print `tl.tensor` using the native python print and check all values of the tensor

Acknowledgement

- The Hatchet team
- Special thanks to Ian Lumsden

Triton Conference 2024

- The Triton Conference is happening again on *September 17th, 2024* in Fremont (CA)
- If you are interested in attending, please fill up this form
  - https://tinyurl.com/4rdfy8s9