McStas / McXtrace instrument simulation

- Ray tracing through components

Components
McStas / McXtrace instrument simulation

- Ray tracing through components
- Port heavy part of component to GPU (Shows some potential)
McStas / McXtrace instrument simulation

- Ray tracing through components
- Port heavy part of component to GPU (Shows some potential)
McStas / McXtrace instrument simulation

- Ray tracing through components
- Port heavy part of component to GPU (Shows some potential)
McStas / McXtrace instrument simulation

- Ray tracing through components
- Port heavy part of component to GPU (Shows some potential)
- Port neutron loop to GPU
  - Lots of code
  - Code generation
  - Compiles! Speed up today

Components
McStas / McXtrace instrument simulation

**CPU MPI 4 cores (95 % usage)**

```
sys: ~:0.26s
ed/cc:0.31s
ed/cc:0.31s
ed/cc:0.31s
ed/cc:0.31s
Simulation 'template_body_simple' (template_simple.instr): running on 4 nodes (master is 'edr32091.dip.tu-dresden.de', MPI version 3.1).
[template_body_simple] Initialize
[template_body_simple] Initialize
[template_body_simple] Initialize
[template_body_simple] Initialize
Save [template_body_simple]
Finally [template_body_simple]: .J. Time: 4:15:46 [h]
Finally [template_body_simple]: .J. Time: 4:15:46 [h]
Finally [template_body_simple]: .J. Time: 4:15:46 [h]
```

16.12 s (Single core 56.0 s)

**GPU 5% usage**

```
run: job 1476679 has been allocated resources
Bash-4.1.5 g++ -ta-tesla.managed,nowlvm -Minfo-accel template_simple3.c -ONOSIGNALS -O3_GCC_RAND_ARG=5°C
bash-4.1.5 cd Hack/
Vishal/ template_simple.c template_simple.instr template_simple.instr
bash-4.1.5 cd Hack/Vishal/ template_simple.c template_simple.instr template_simple.instr
bash-4.1.5 is a.out mc_code1 mc_code2.c template_simple3.c template_simple3.c template_simple3.c template_simple3.c template_simple3.c
bash-4.1.5 time ./a.out Part1
[template_body_simple] Initialize
Save [template_body_simple]
Finally [template_body_simple]: .J. Time: 4:15:46 [h]
```

5.43 s
What problems you encountered

• Problems with legacy app structure. Complex, nested loops, static and extern variables, printf

• Issues with algorithm. Not-so-high modularity did not really encourage parallelization

• Tool bugs

• Tool lack of features

• System setup Code-Gen vs. .c instument
Wishlist

• What do you wish existed to make your life easier?
  • Tools
  • Language standards
  • Event
    *More GPU Hackathons!*
  • Systems
Was it worth it?

- Was this worth it?
  - Me
    - openACC and GPU experience
  - Your team
    - openACC experience, some CUDA!
- Your app
  - Parallelization compilation and runtime works for a simple (although full) instrument.
- Your domain
  - The insight gained can be shared with the main McXtrace/McStas development
- Your center
- Your mentors
  - Simply amazing support
    - Will you continue development? Of course!