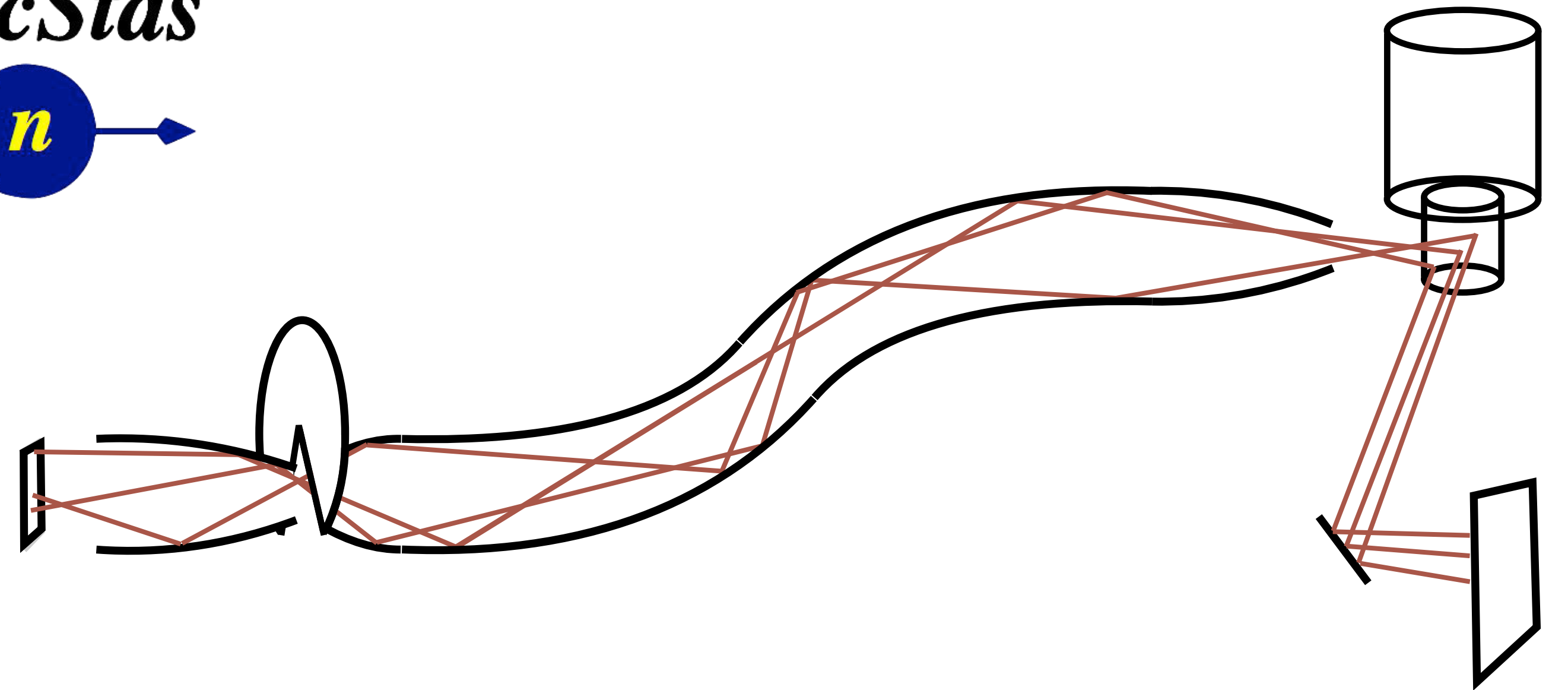


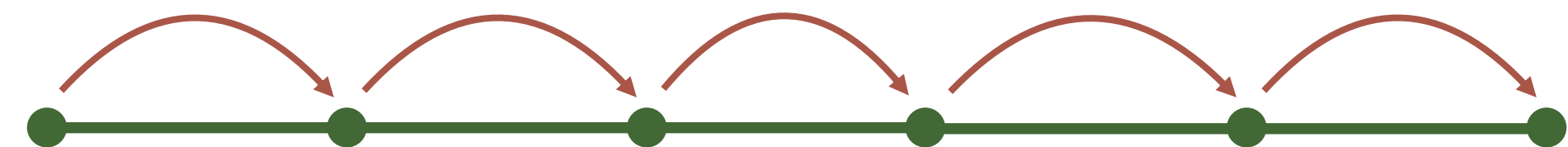
# McStas / McXtrace instrument simulation

- Ray tracing through components

***McStas***



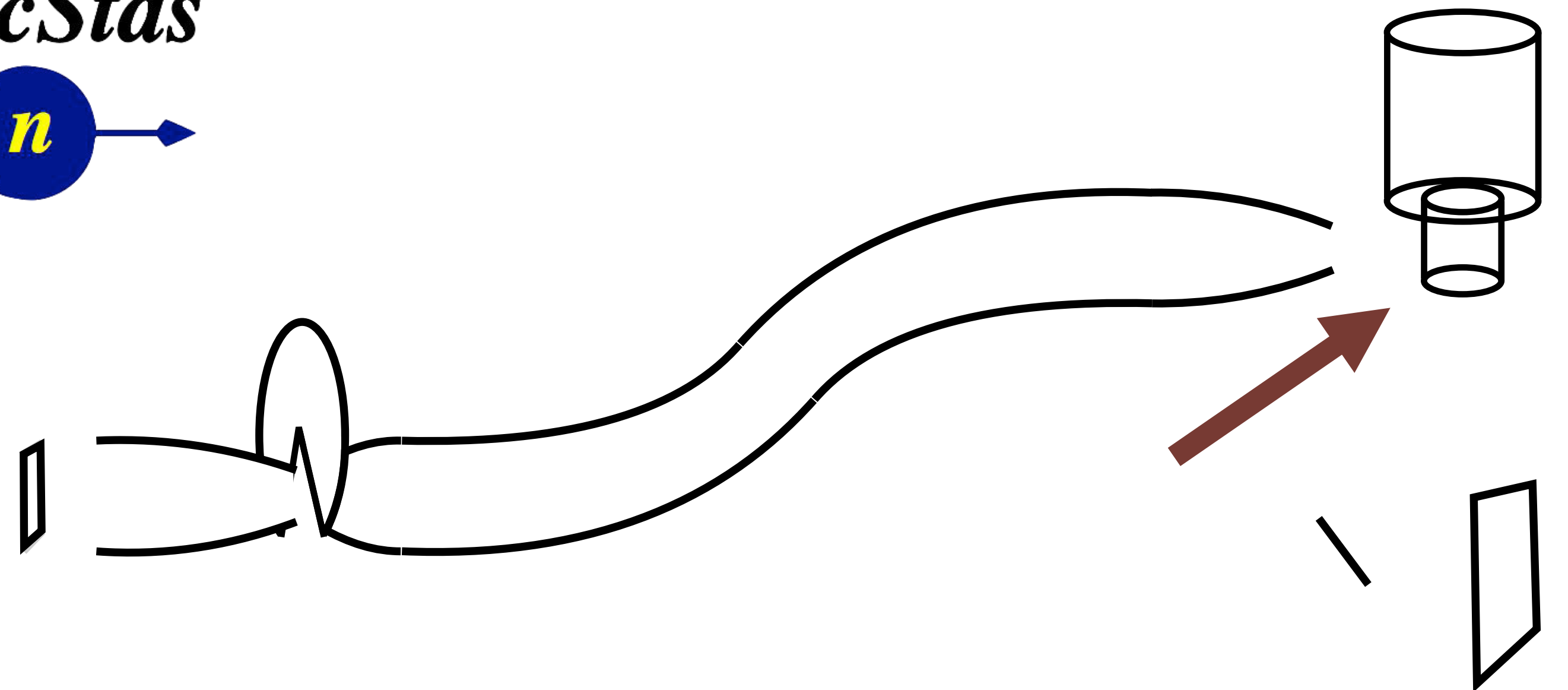
Components



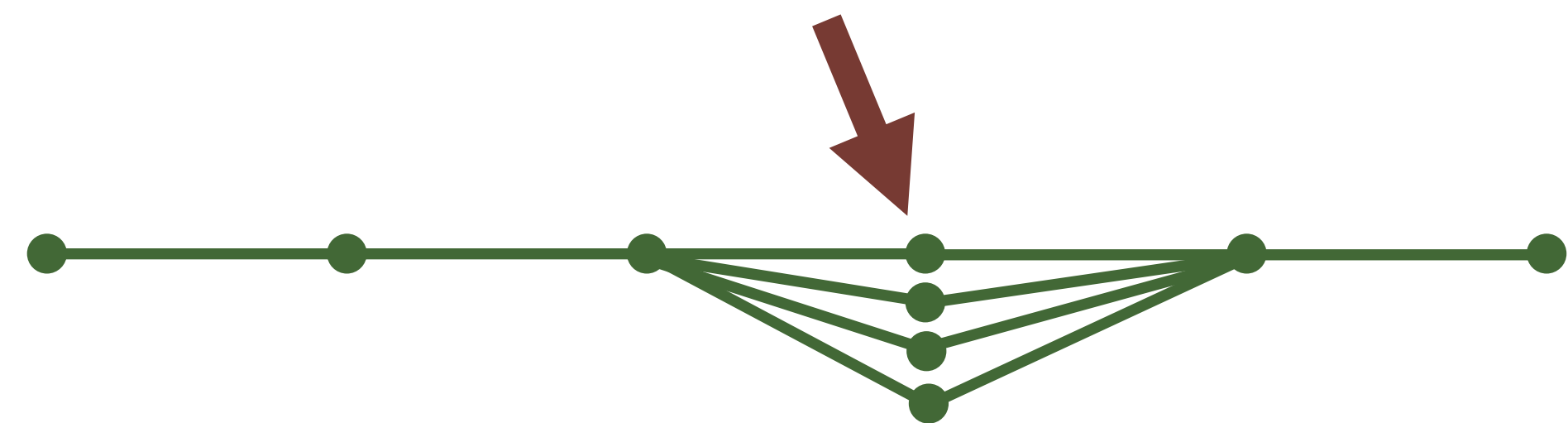
# McStas / McXtrace instrument simulation

- Ray tracing through components
- Port heavy part of component to GPU (Shows some potential)

***McStas***

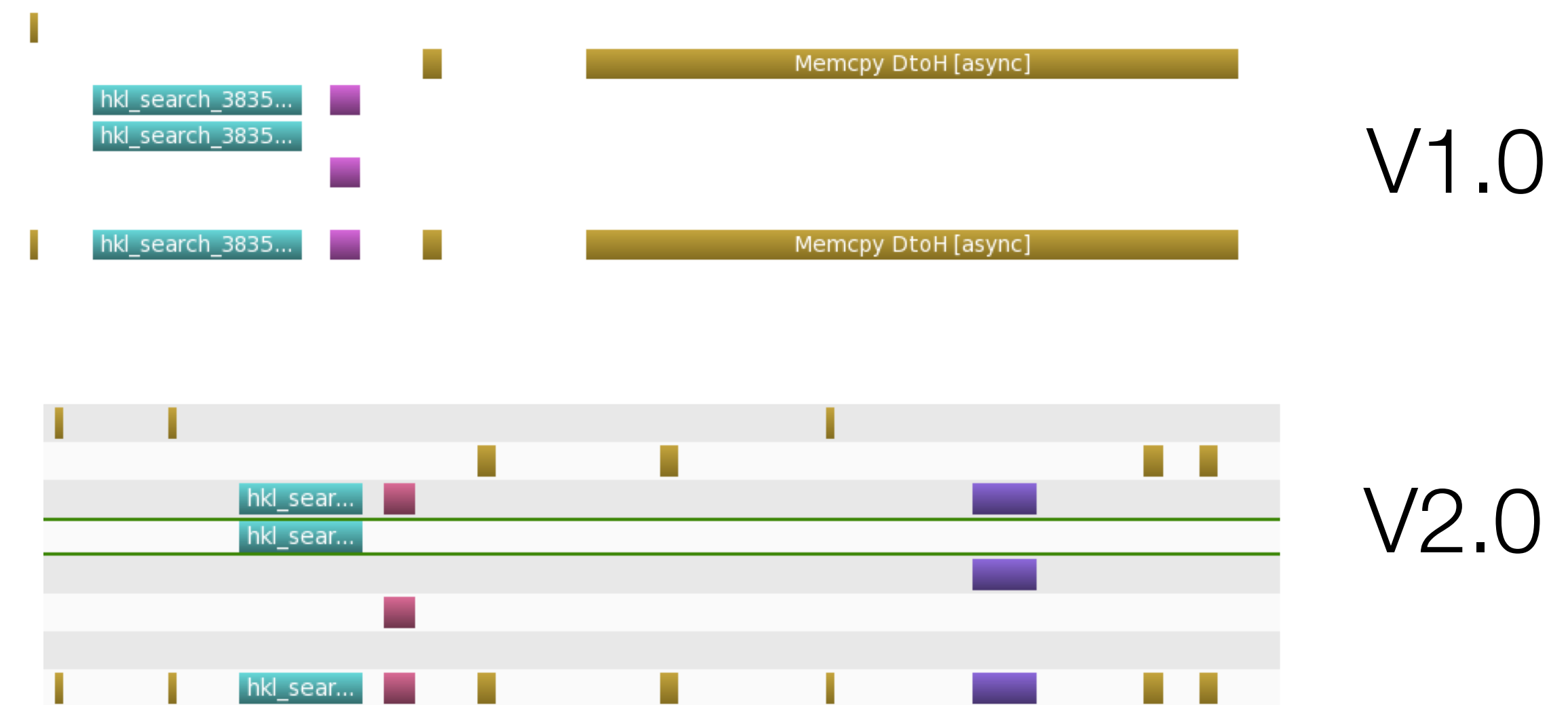


Components

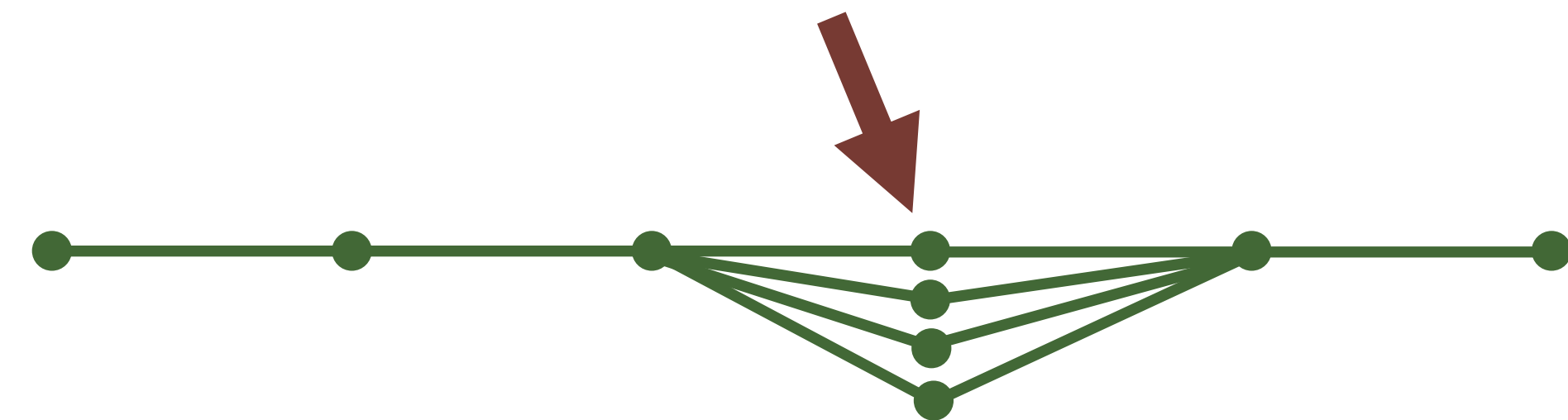


# McStas / McXtrace instrument simulation

- Ray tracing through components
- Port heavy part of component to GPU (Shows some potential)

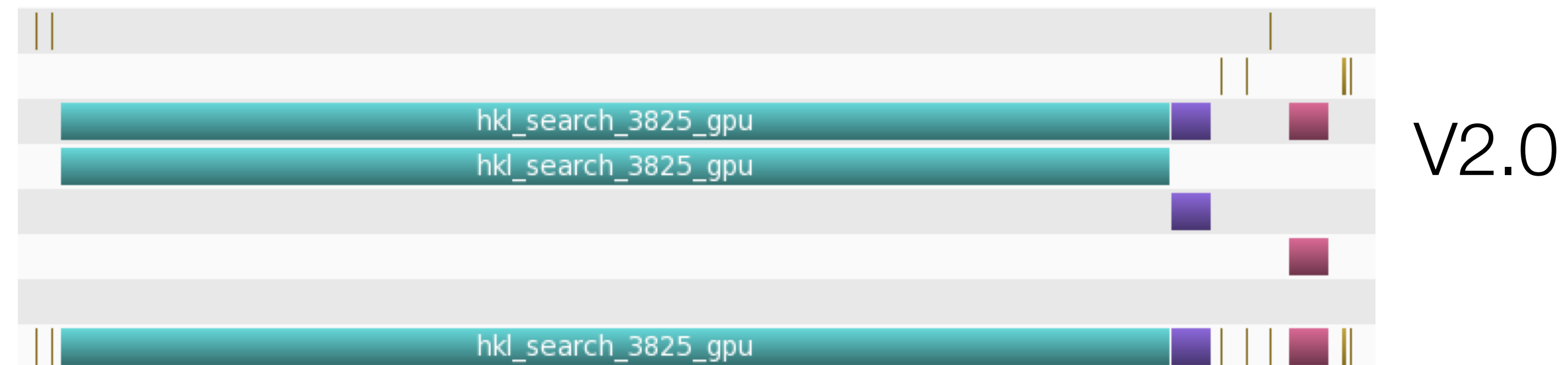


## Components

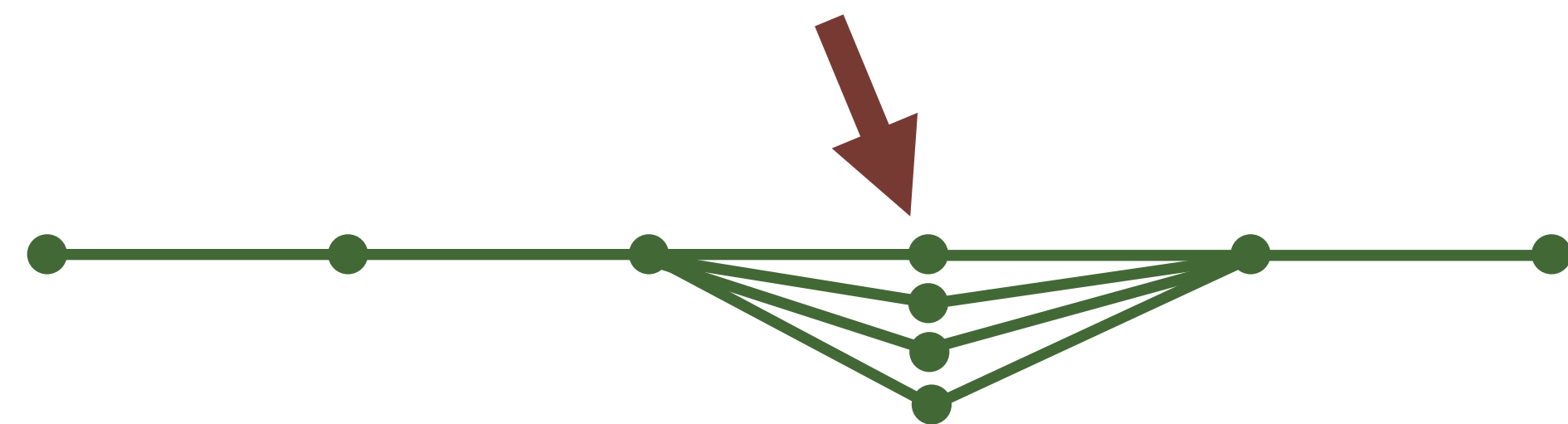


# McStas / McXtrace instrument simulation

- Ray tracing through components
- Port heavy part of component to GPU (Shows some potential)



## Components



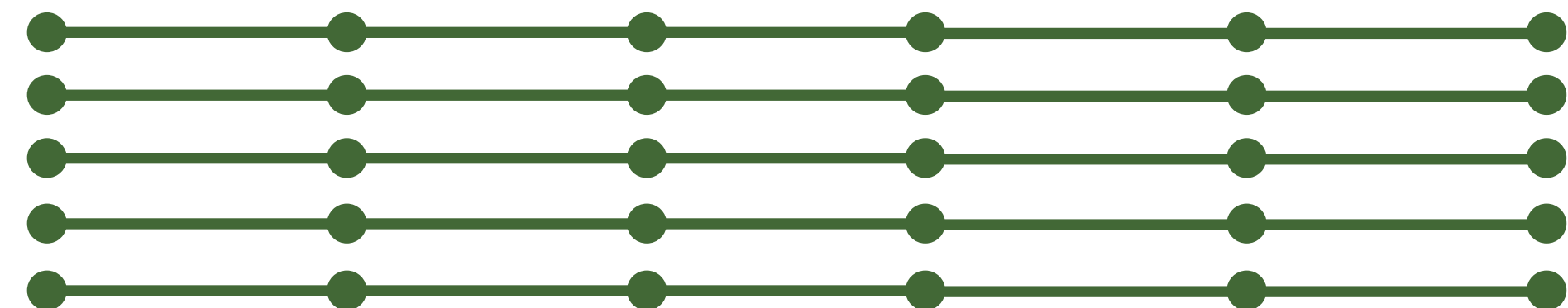
# McStas / McXtrace instrument simulation

- 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

***McStas***



Components





# McStas / McXtrace instrument simulation

CPU MPI 4 cores (95 % usage)

```
Friday — -bash — 116x24
sys      0m0.260s
edr32091:Friday pkwi$
edr32091:Friday pkwi$ time mpirun -np 4 ./a.out -n1e9 Par1=1
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: .]. Time: 415460 [h]

Finally [template_body_simple: .]. Time: 415460 [h]

Finally [template_body_simple: .]. Time: 415460 [h]

Finally [template_body_simple: .]. Time: 415460 [h]

real    0m16.126s
user    1m3.158s
sys      0m0.208s
edr32091:Friday pkwi$
```

16.12 s (Single core 56.0 s)

GPU 5% usage

```
pkwi — ssh taurus.campus — 108x24
srun: job 14706670 has been allocated resources
[bash-4.1$ pgcc -ta=tesla,managed,nollvm -Minfo=accel template_simple3.c -DNOSIGNALS -DMC RAND_ALG=5^C
[bash-4.1$ time ./a^C
bash-4.1$ pwd
/home/gpu29/McCode/Dresden-intermediate-results
bash-4.1$ cd
bash-4.1$ cd Hack/
[Vishal/      template_simple.c      template_simple.instr  template_simple.instr~
bash-4.1$ cd Hack/
[Vishal/      template_simple.c      template_simple.instr  template_simple.instr~
bash-4.1$ cd Hack/Vishal/
bash-4.1$ ls
a.out mccode.sim template_simple2.c~ template_simple3.c  template_simple3.c~
bash-4.1$ time ./a.out Par1=1 -n1e9
[template_body_simple] Initialize

Save [template_body_simple]

Finally [template_body_simple: .]. Time: 415460 [h]

real    0m5.431s
user    0m3.956s
sys      0m1.428s
bash-4.1$
```

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 instrument

# 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!