OAK RIDGE National Laboratory

Scaling Full-system Simulation of ARM SVE Processors Using Compilers and Runtime Tool APIs

Matthew Baker
Jeffrey Young
Oscar Hernandez

September 17, 2019

Oak Ridge National Laboratory

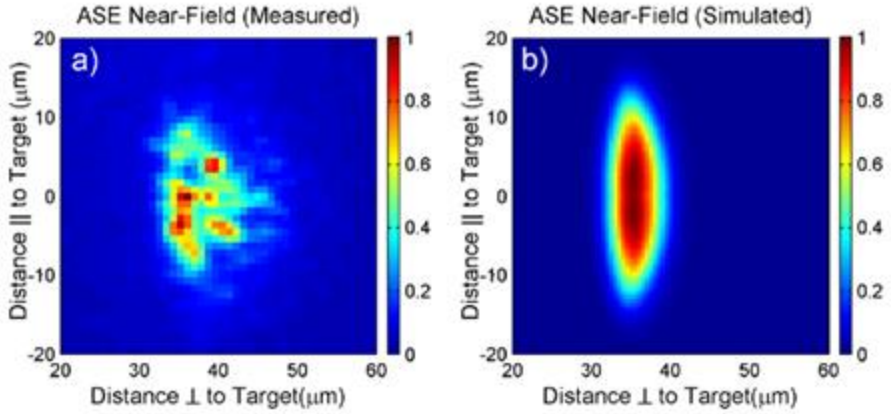
ORNL is managed by UT-Battelle, LLC for the US Department of Energy

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725

U.S. DEPARTMENT OF ENERGY

Improving Co-design of Software and Hardware

- Software
  - The DoE uses mini-apps to represent workloads of interest, many using OpenMP or CUDA
- Hardware
  - Can we use simulation to model future hardware and architectures at scale?
- We also want to investigate how OpenMP mini-apps might be vectorized for future SVE or vector-based hardware
  - XRayTrace, XGC, EPCC, etc.



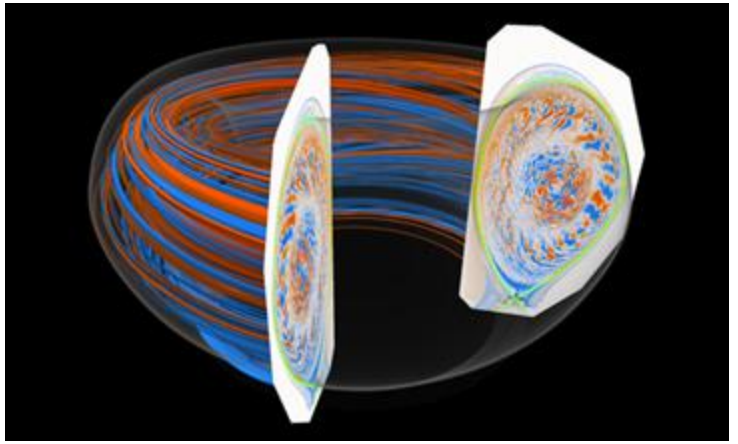


Photo sources: C.-S. Chang, Present and Future Computing Requirements for Tokamak Edge Physics https://www.nersc.gov/assets/Uploads/CSChang.pdf
M. Berrill, XRayTrace Miniapp presentation, 2017

Current Problems for Co-Design

- It is impractical to simulate an entire program; we must pick *regions of interest* (ROI)
- However, it is much more challenging to programmatically pick regions of interest.
  - Many current techniques are intrusive (i.e., manual pragma insertion) and are not conducive to reproducibility
  - Examples:
    - Gem5 manual ROI and checkpointing, ARMIE ROI for memory tracing

```
[bzf@wombat-login2 gem5-tools]$ cat checkpoint.c
#include <gem5ops.h>

int main(int argc, char** argv)
{
        m5_checkpoint(0,0);
        return 0;
}
[bzf@wombat-login2 gem5-tools]$ armie -msve-vector-bits=2048 --iclient libinscount_emulated.so ./check
point
Client inscount is running
261312 instructions executed of which 0 were emulated instructions

Guest process terminated by signal: 15687 Illegal instruction ./checkpoint
[bzf@wombat-login2 gem5-tools]$
```

How do we make codesign easier and more scalable?

- Use tooling to identify and simulate regions of interest
  - OpenMP Tools (OMPT) interface to instrument applications at runtime
  - Handle incompatibilities between tools like gem5 and ARMIE
    - ARMIE is unhappy when it runs into gem5 magic instructions!
- Provide a consistent runtime between simulation environments
  - Use containers to bring user defined software stacks across tools
  - Specifically, use Charliecloud to deploy tool-friendly mini-apps for ARMIE and gem5
- Combine OMPT techniques with CharlieCloud to provide isolation and easy of deployment
  - Different OMP tools can be built as a separate libraries and then run with whatever OMP 5.0 runtime is part of a disk image or container

OMPT for runtime simulator hooks

- OMPT is a new tool interface for OpenMP 5.0+
- Allows tool developers to add analysis and introspection to the OpenMP runtime
  - Mini-app annotations can be added in a non-intrusive way
  - No need to manually instrument code with gem5 ops or ARMIE ROI pragmas; OMPT callbacks can be used to add annotations

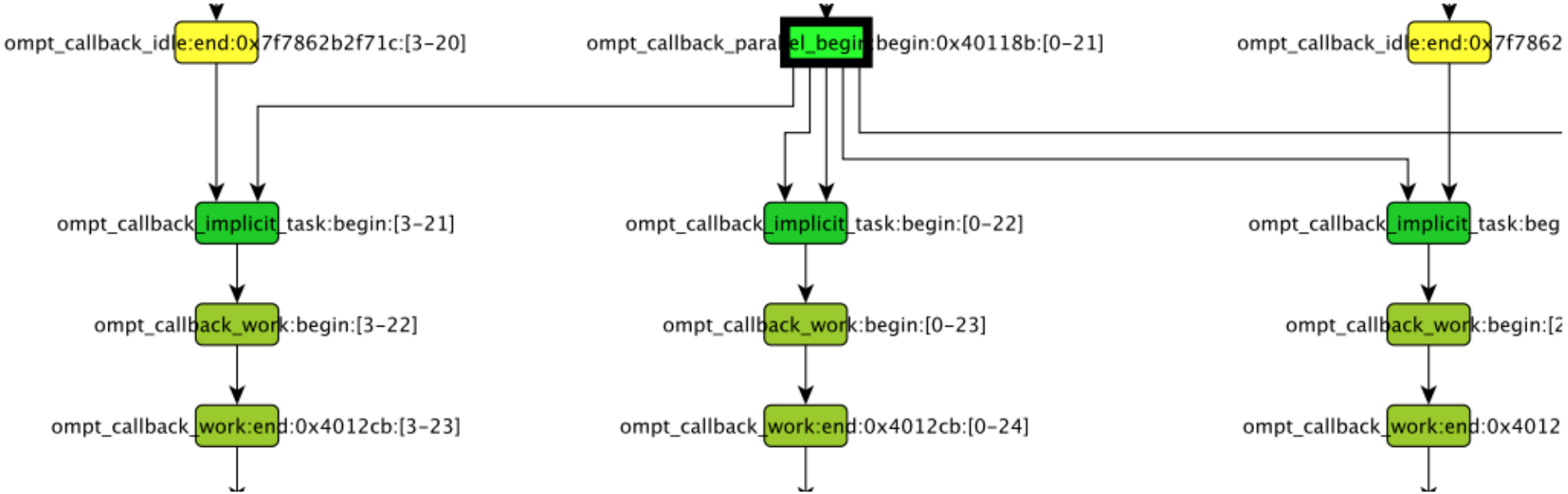
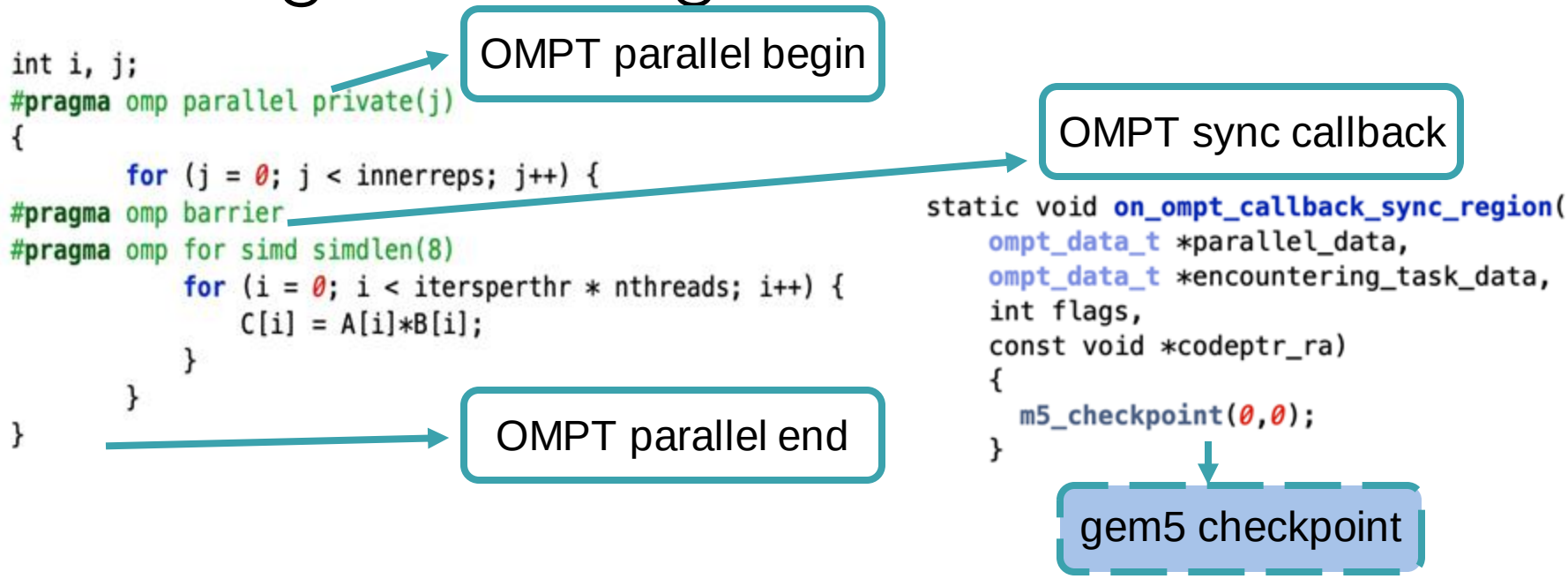


Image from Yonghong Yan, Philip Conrad, Yudong Sun, "Visualizing OpenMP Execution using OMPT". SC 2018

Extending OMPT for gem5 Simulations

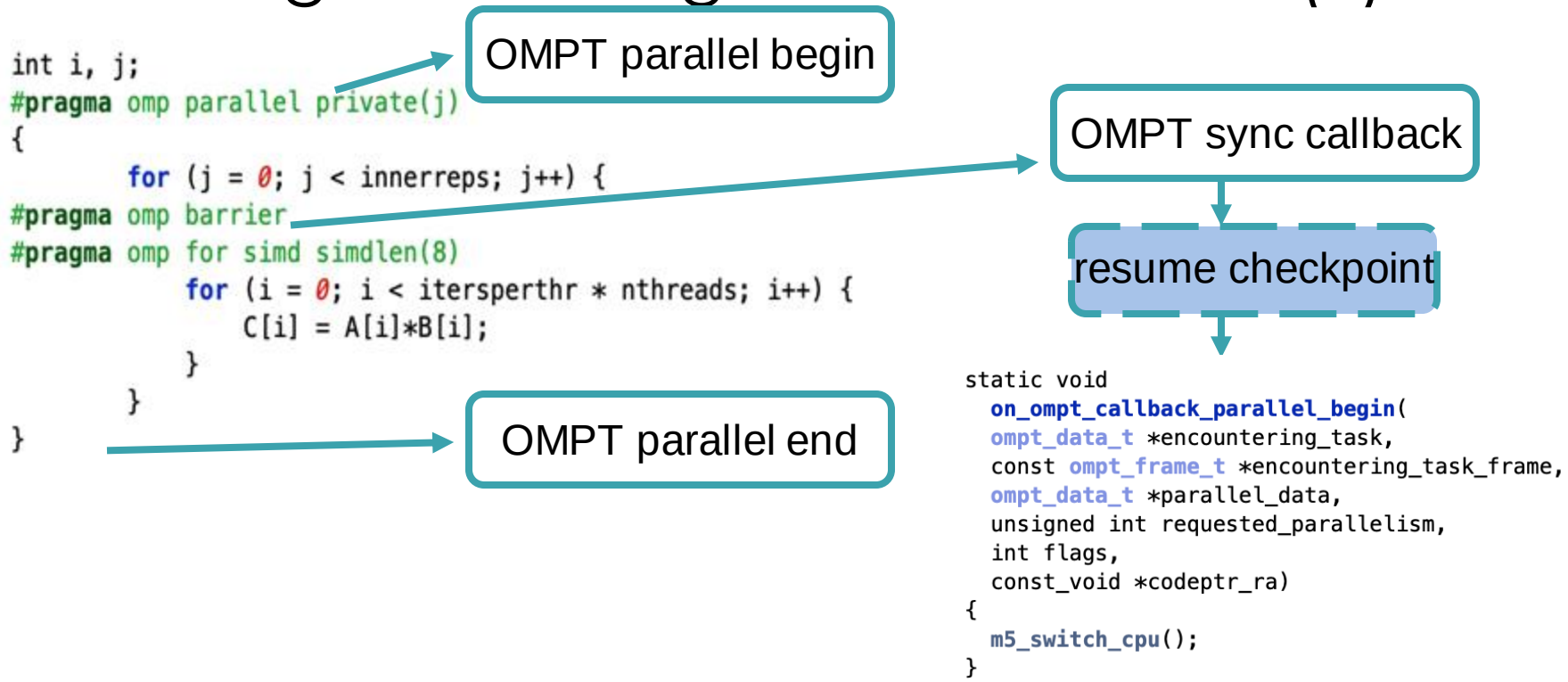



```
int i, j;
#pragma omp parallel private(j)
{
        for (j = 0; j < innerreps; j++) {
#pragma omp barrier
#pragma omp for simd simdlen(8)
            for (i = 0; i < itersperthr * nthreads; i++) {
                C[i] = A[i]*B[i];
            }
        }
}
```

```
static void on_ompt_callback_sync_region(
    ompt_data_t *parallel_data,
    ompt_data_t *encountering_task_data,
    int flags,
    const void *codeptr_ra)
{
  m5_checkpoint(0,0);
}
```

- Initial work has used the OpenMP Tools API to drop gem5 checkpoints with simulators like gem5
- OMPT callbacks can be currently used to track *omp_parallel, omp_barrier, omp_single*

Extending OMPT for gem5 Simulations (2)



- Initial work has used the OpenMP Tools API to drop gem5 checkpoints with simulators like gem5
- OMPT callbacks can be currently used to track *omp_parallel, omp_barrier, omp_single*

Creating a Consistent environment For Simulations

- Ensuring consistent behavior of mini-apps between environments
- Simulators often have their own quirks that need to be managed
- Leverage containers to ensure consistent runtime environment
- Applications to run in simulation may be bundled as containers
  - Easier to distribute
  - Same binary between simulators
  - Reproducible

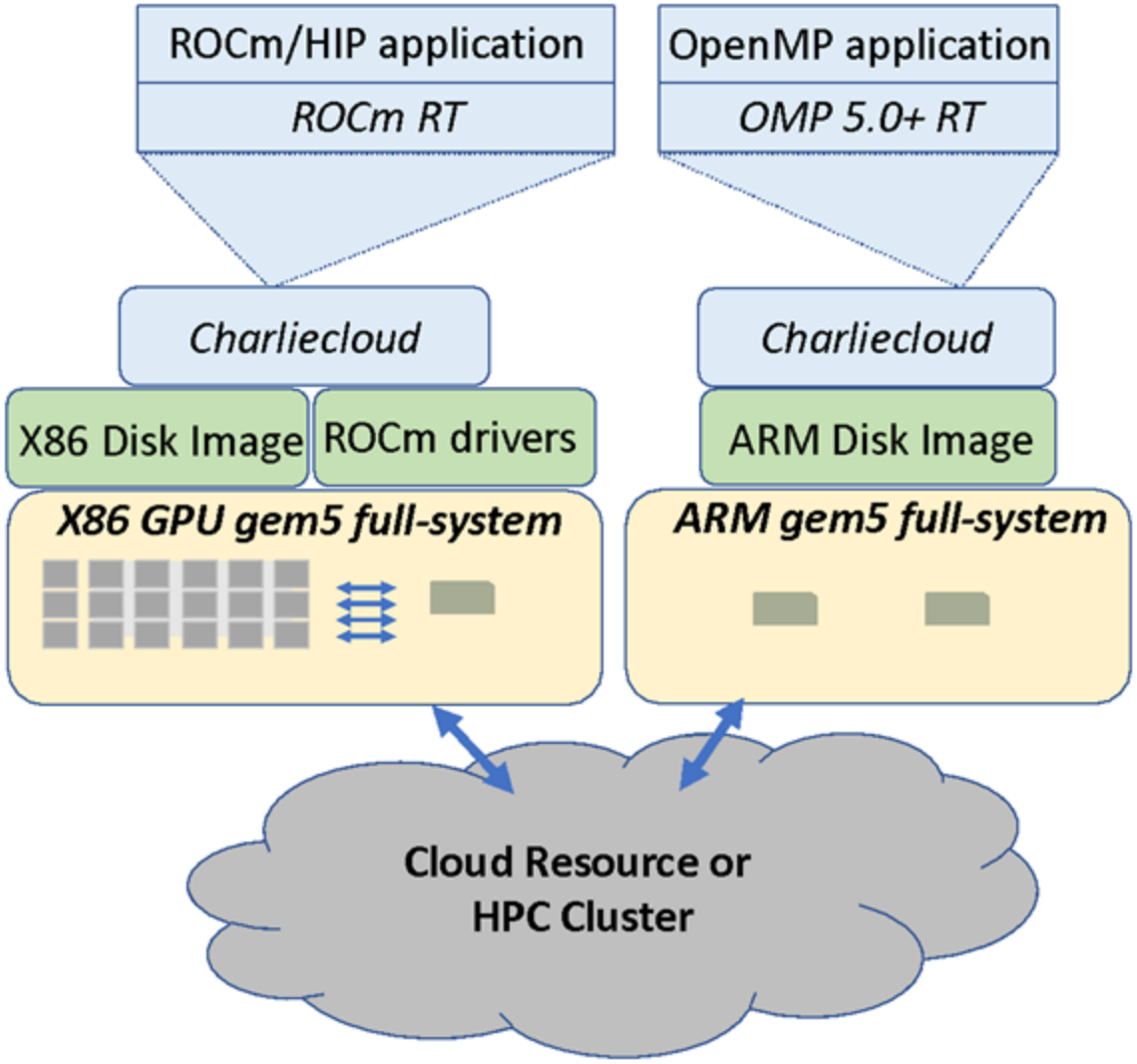
Why Charliecloud?

- While most container environments have heavy requirements and root privileges, Charliecloud does not
  - Simple to include in gem5 full system simulation or run in HPC centers
    - Just need its ch-run binary!
  - Container image flattened into tarball
    - Extract into new directory and container environment is a ch-run away
- Host environment tools can be leveraged with bind mounts
  - ARMIE home dir can be bind mounted into container directory
    - ARMIE tools are then available inside isolated, user defined environment
  - gem5 can bind gem5 OMPT enabled tool directory into simulations

```
[bzf@wombat-login2 slides]$ tar xf simdbench.tar.gz -C simdbench
[bzf@wombat-login2 slides]$ ch-run --bind $ARMIE_PATH simdbench -- /mnt/0/bin64/armie -msve-vector-bit
s=2048 --iclient libinscount_emulated.so /app/simdbench
Client inscount is running
SVE: 0x0000000000401688 0x043f57df
```

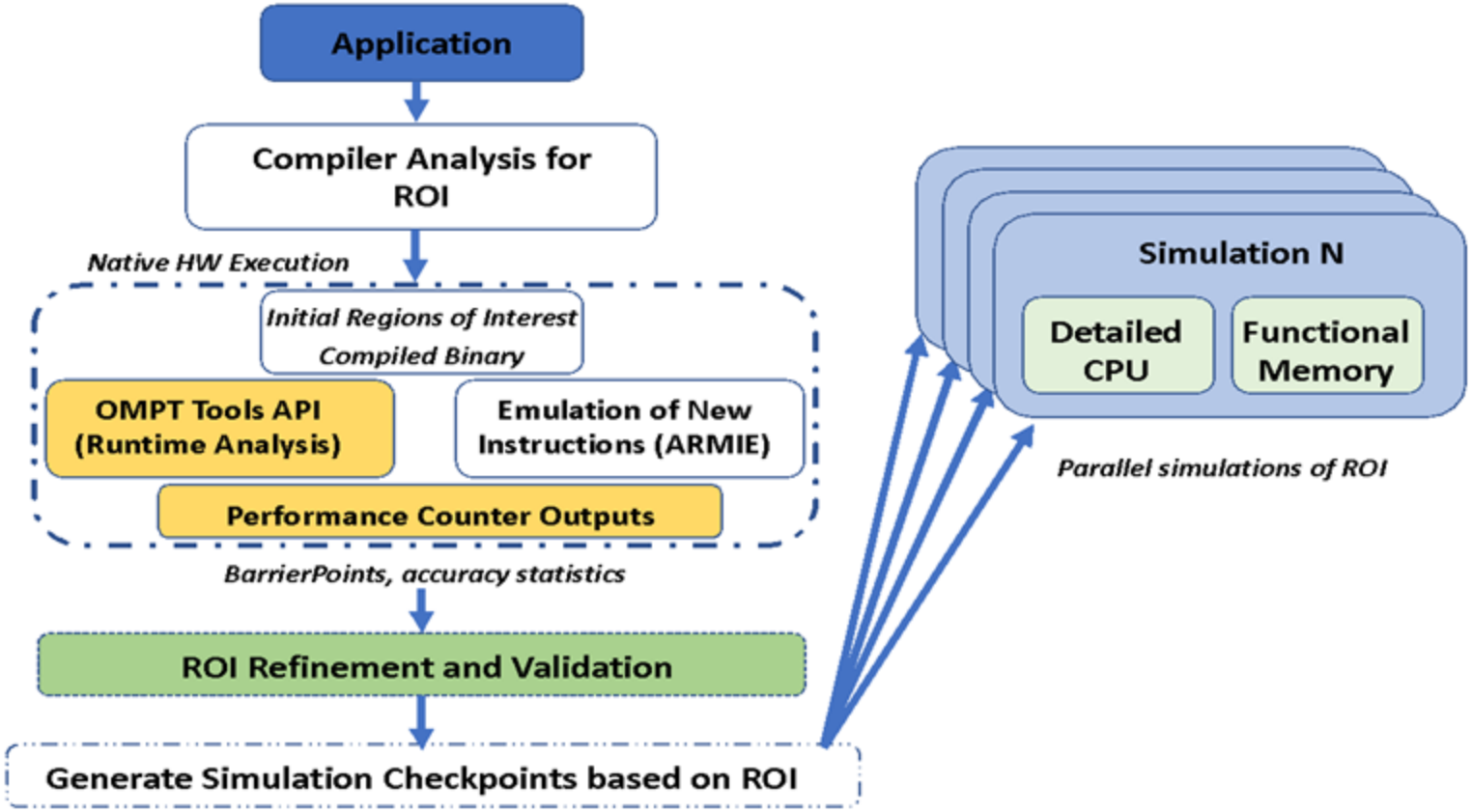
Deployment of Design Space Exploration at Scale

- Allow users to create their own lightweight simulation containers inside FS simulations
- Mirrors current use-cases for systems like Summit (Docker, Singularity) and reduces deployment overheads



Future Work

- Integrate OMPT techniques with BarrierPoint work
- Develop additional OMPT based tools for improving simulation execution and instrumentation



Based on work by Miguel Tairum Cruz, Sascha Bischoff, Roxana Rusitoru, "Shifting the Barrier: Extending the Boundaries of the BarrierPoint Methodology". ISPASS 2018

Acknowledgments

- This work was funded by an Oak Ridge National Laboratory Directed Research & Development (LDRD) project
- This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725.

Backup

What is a Barrier Point?

```
#pragma omp parallel for
for (j = 0; j < innerreps; j++)
{
  #pragma omp barrier
  for (i = 0; i < itersperthr * nthreads; i++)
  {
    C[i] = A[i]*B[i];
  }
}
```

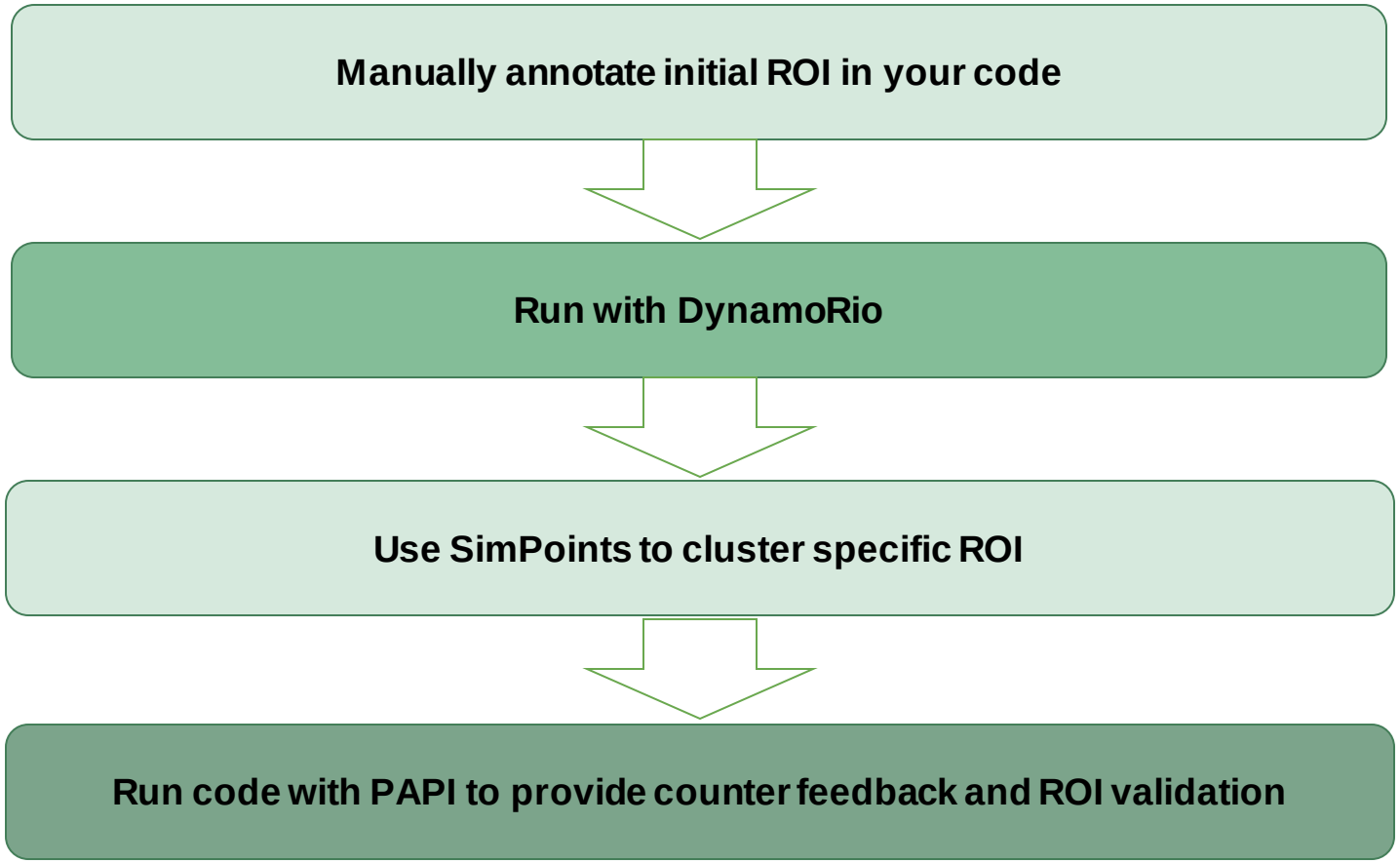
Synchronization Point

Synchronization Point

Synchronization Point

Based on work by Carlson, Trevor E., Wim Heirman, Kenzo Van Craeynest, and Lieven Eeckhout. "Barrierpoint: Sampled simulation of multi-threaded applications." ISPASS 2014

Barrier Points Workflow



Based on work by Miguel Tairum Cruz, Sascha Bischoff, Roxana Rusitoru,
"Shifting the Barrier: Extending the Boundaries of the BarrierPoint Methodology". ISPASS 2018

Why is the concept of BarrierPoints relevant for architectural simulation and OMPT?

- Gem5 has to be run with parallel configurations in an embarrassingly parallel fashion
- OMPT helps! it's a good way to extend and implement new BarrierPoint-like features
  - We envision that OMPT could be be used to automatically add regions of interest as well as trigger some of the analysis BarrierPoints uses (e.g., trigger DynamoRio emulation or PAPI counters)