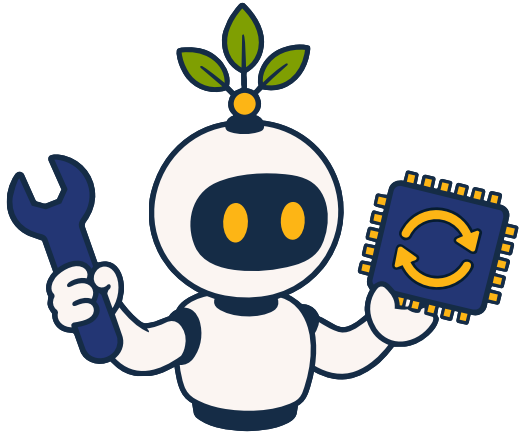


CHIA

https://chialoops.ai

Berkeley Architecture Research

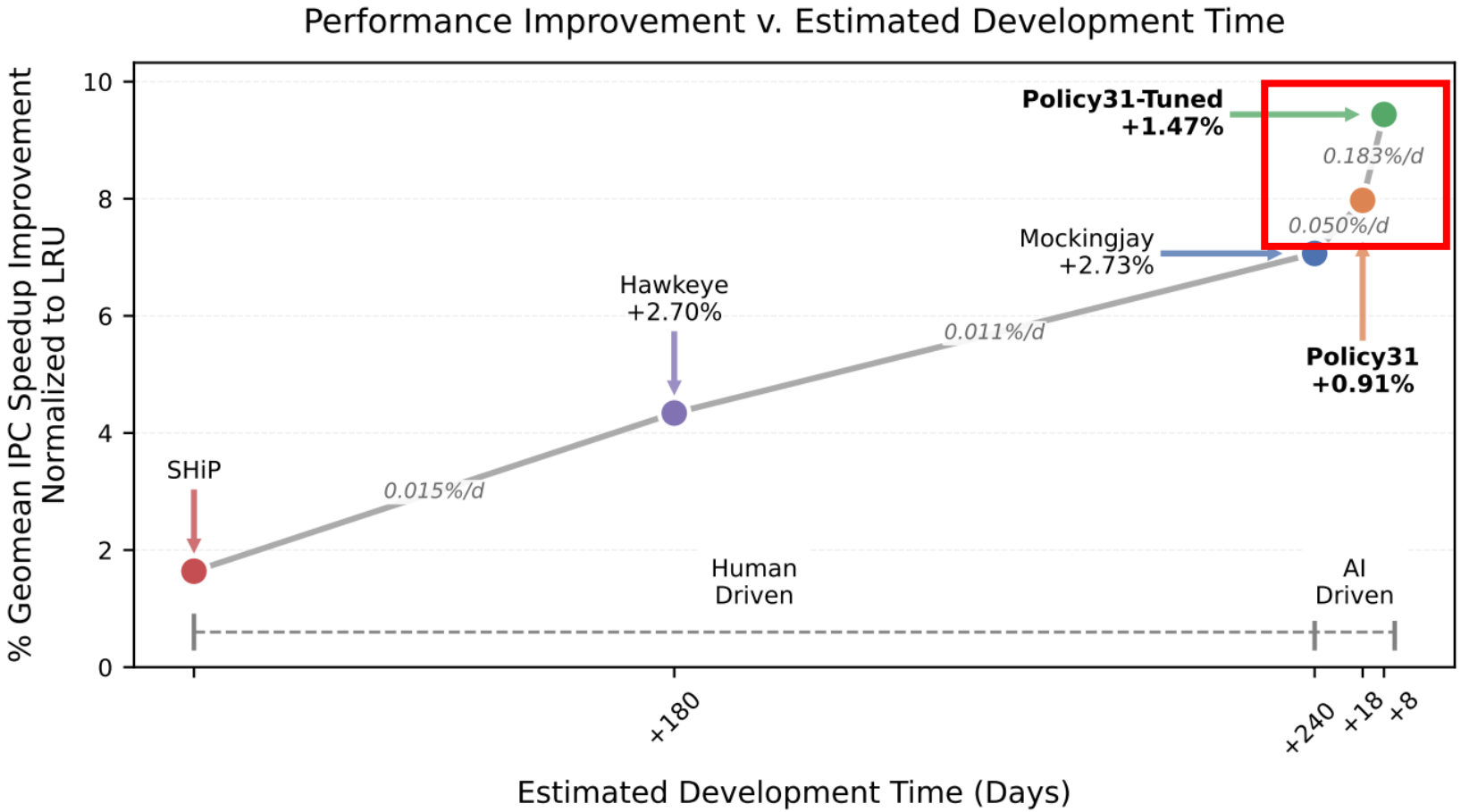
CHIA: An open-source framework for principled, agentic AI-driven hardware/software co-design research

MLArchSys, ISCA 2026

Angela Cui*, Ferran Hermida-Rivera*, Jack Toubes*,
Raghav Gupta, Jim Fang, Chengyi Lux Zhang, Ella Schwarz, Junha Kim,
Yakun Sophia Shao, Borivoje Nikolić, Christopher W. Fletcher, Sagar Karandikar

*Equal Contribution

AI-driven design brings huge opportunities



Use AlphaEvolve to agentically discover new cache replacement policies

Up to 12x speedup in (Perf improvement) / (Dev time) compared to human

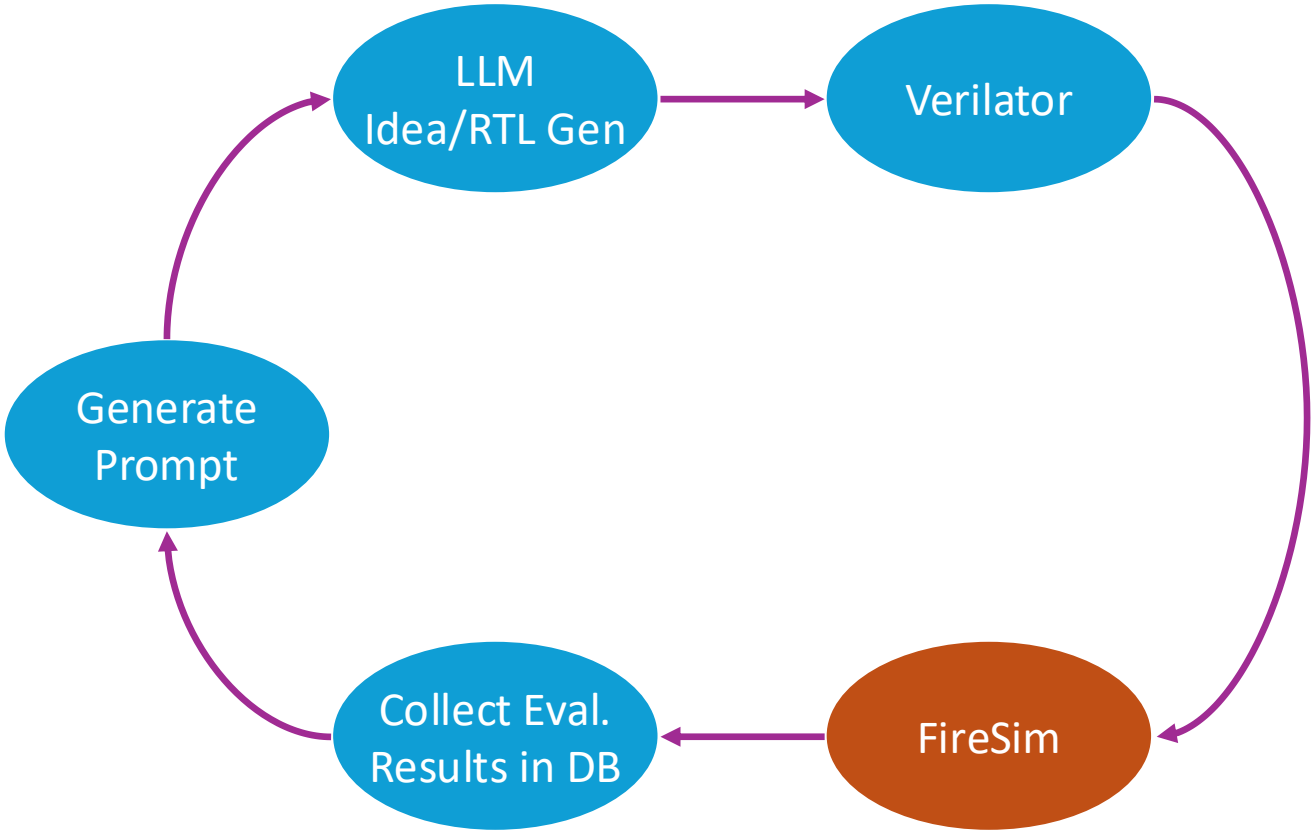
Raghav Gupta, Akanksha Jain, Abraham Gonzalez, Alexander Novikov, Po-Sen Huang, Matej Balog, Marvin Eisenberger, Sergey Shirobokov, Ngân Vu, Martin Dixon, Borivoje Nikolić, Parthasarathy Ranganathan, and Sagar Karandikar. 2025. **ArchAgent: Agentic AI-driven Computer Architecture Discovery.** https://openreview.net/forum?id=hcxN9l6zqZ

Challenges

- The whole point of applying AI: We want the computers to be doing the work for us
  - High-level uarch sim won't cut it as the evaluation endpoint
- Many proposed evolutionary flows/algorithms
  - Algorithmic differences are interesting
  - But the flows are static; designed for a specific use-case
- Immense engineering/"good science" challenges for long-running/highly parallel evals (i.e., all of HW/SW co-design)
  - Agentic flows work best with evaluators that are verified

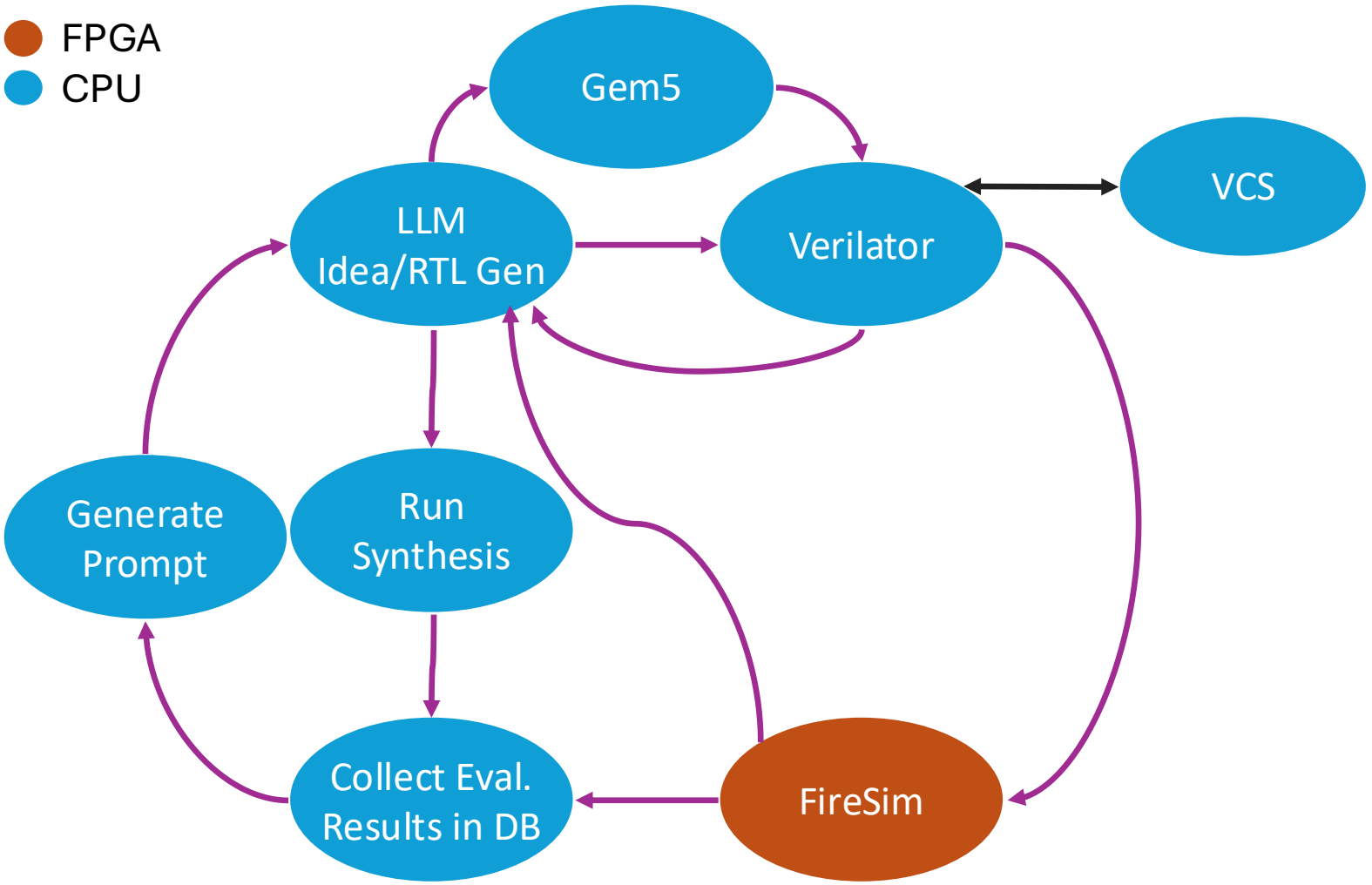
A basic specification -> RTL agentic loop

FPGA
CPU



Maybe a script automates the steps of this loop…

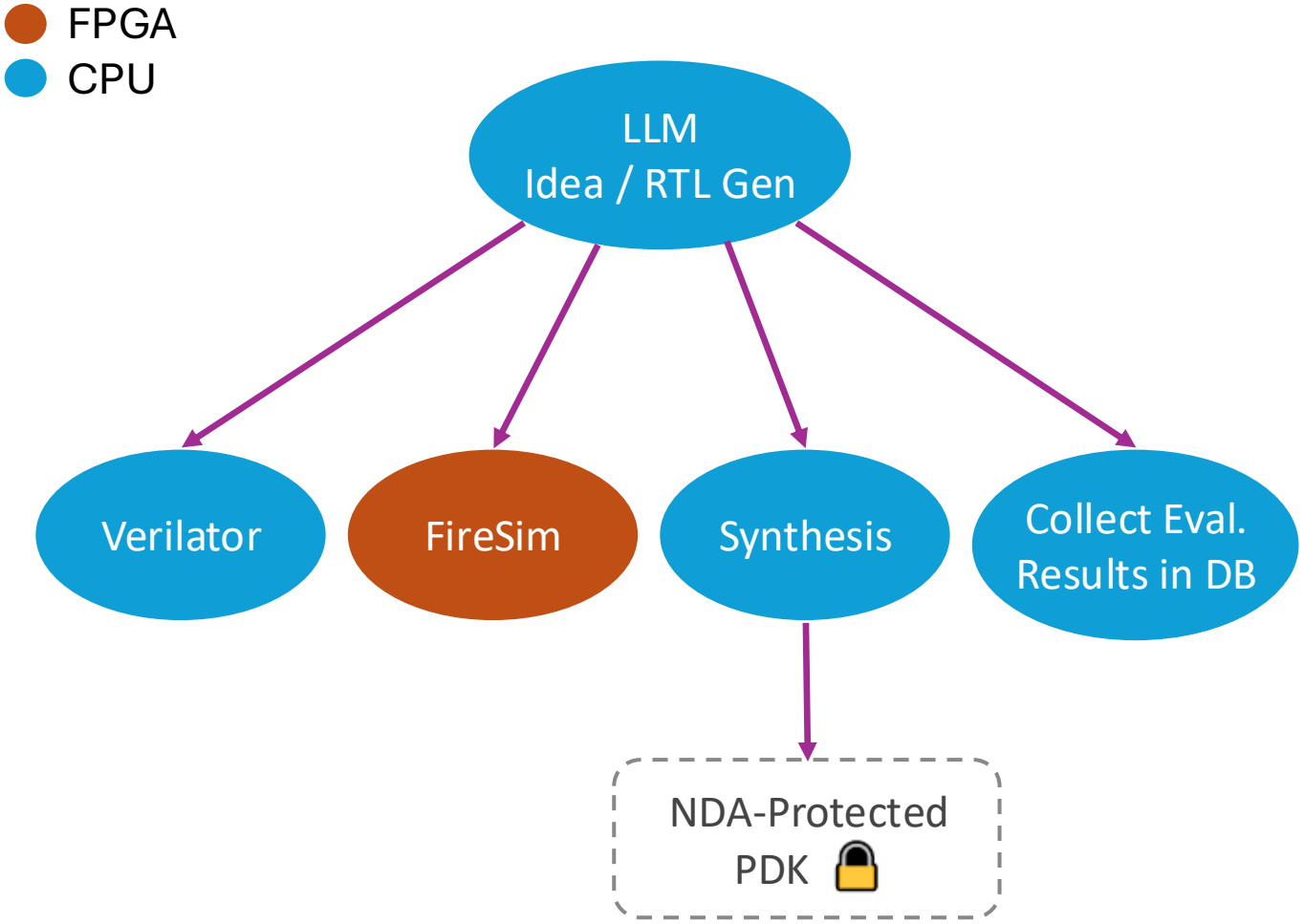
What if we want to tune the flow?



What the framework needs

- **Enables agile experimentation**

Don’t let the agent run wild

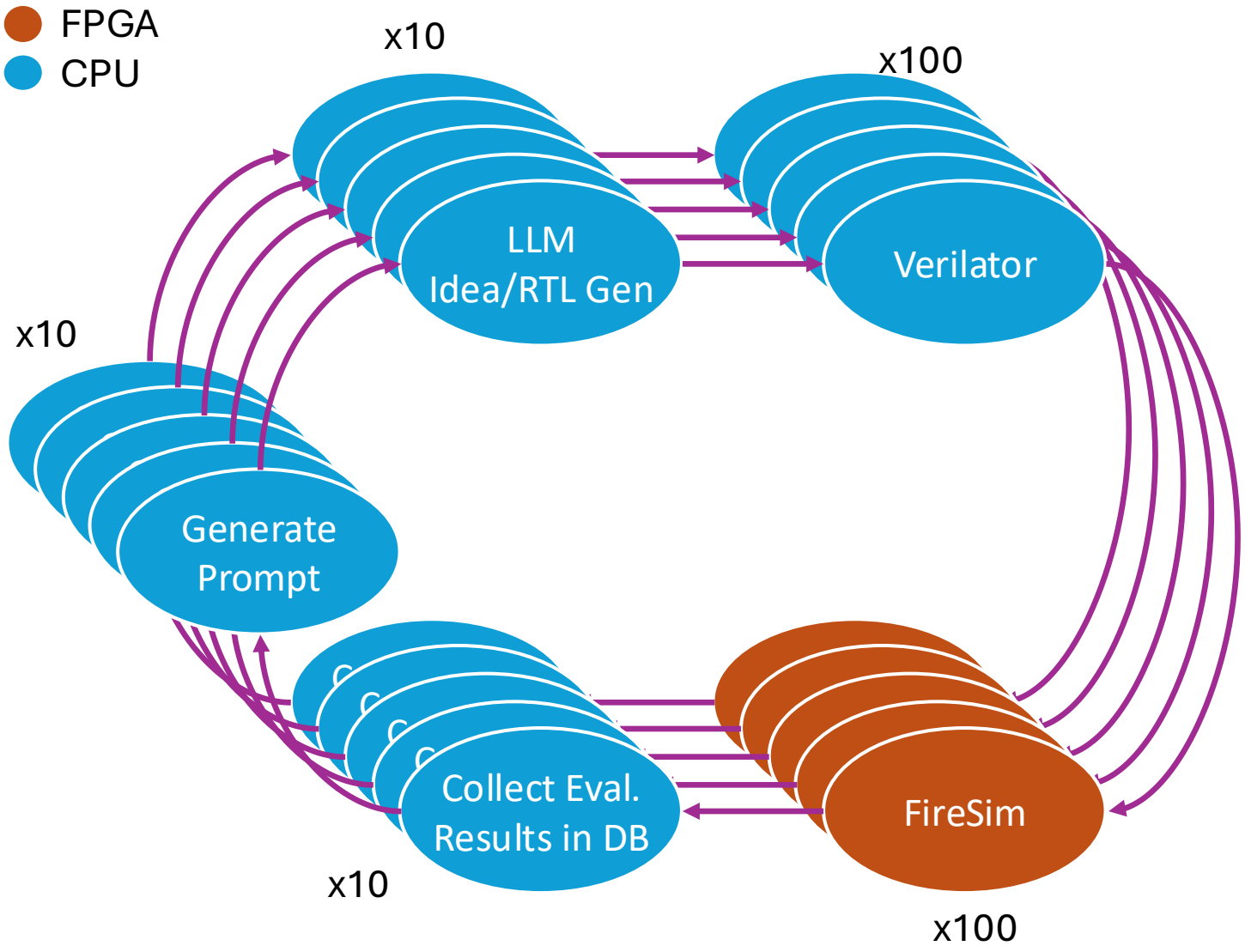


What the framework needs

✓ Enables agile experimentation

▸ **Control flow flexibility + expressiveness**

LLMs can do 10 things at once



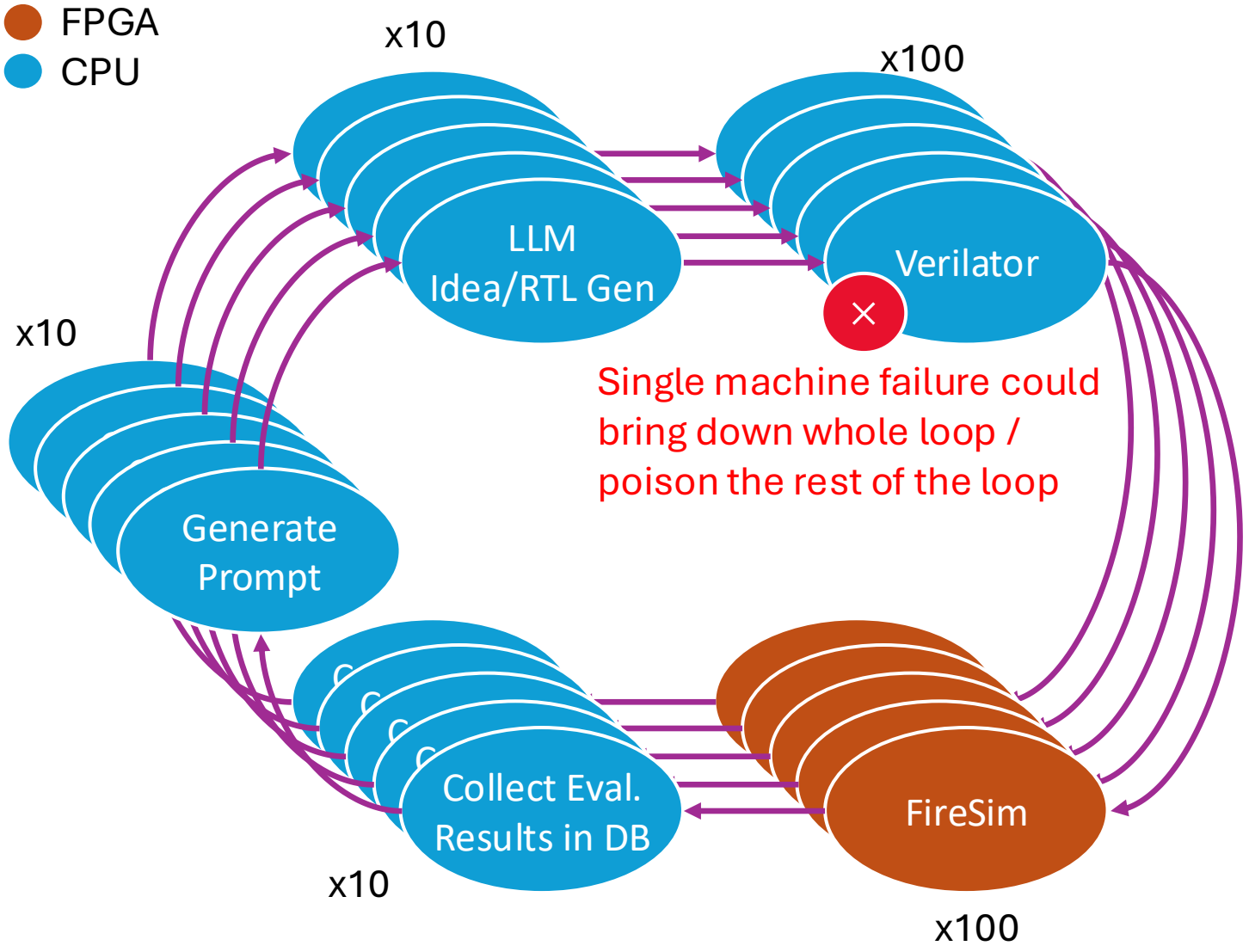
What the framework needs

✓ Enables agile experimentation

✓ Control flow flexibility + expressiveness

- **Parallelization -> Distribution**

Failures become a significant concern at scale



What the framework needs

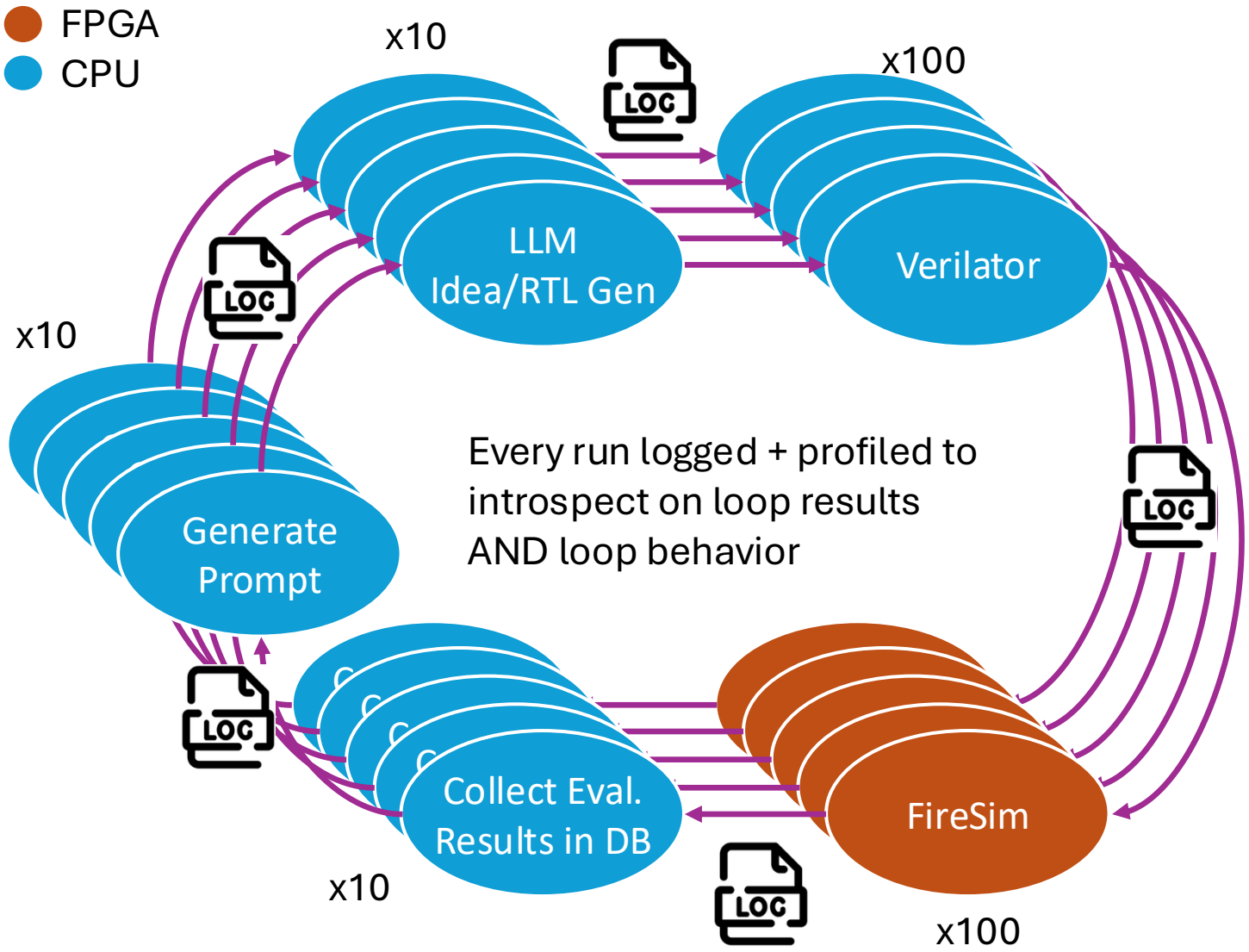
✓ Enables agile experimentation

✓ Control flow flexibility + expressiveness

✓ Parallelization -> Distribution

▸ **Fault tolerance**

Do good science with data collection



What the framework needs

✓ Enables agile experimentation

✓ Control flow flexibility + expressiveness

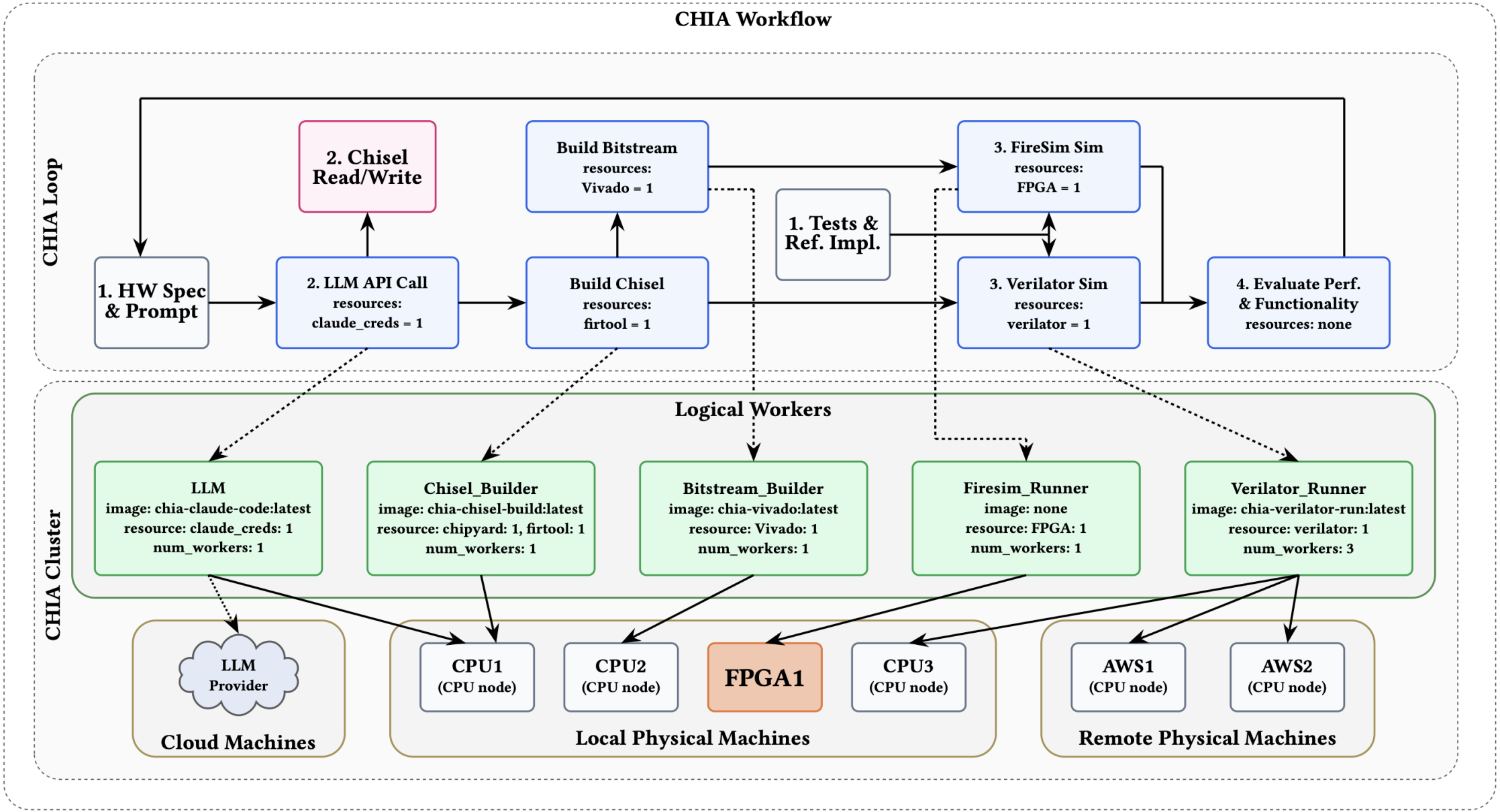
✓ Parallelization -> Distribution

✓ Fault tolerance

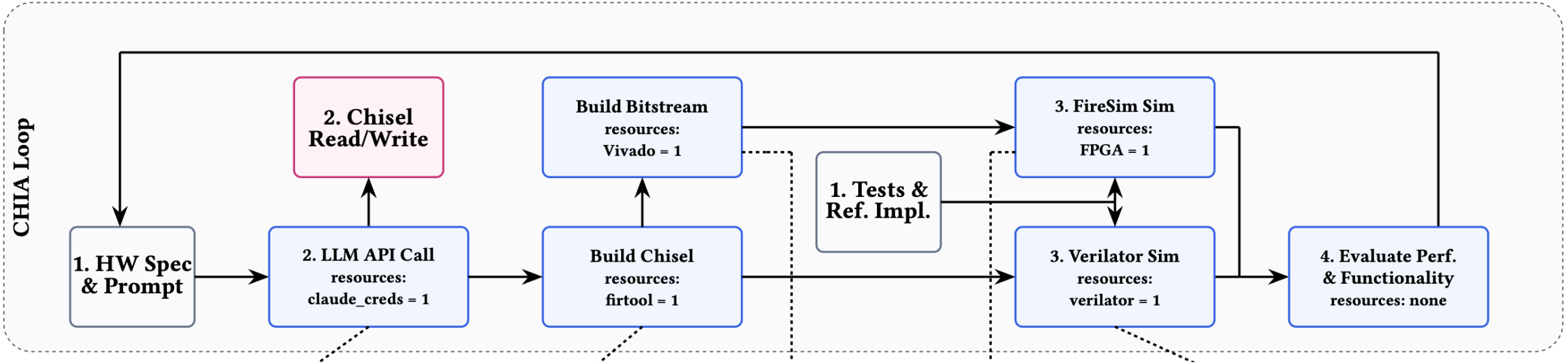
▸ **Data collection, profiling, introspection**

Chia Design and Building a Chia Flow

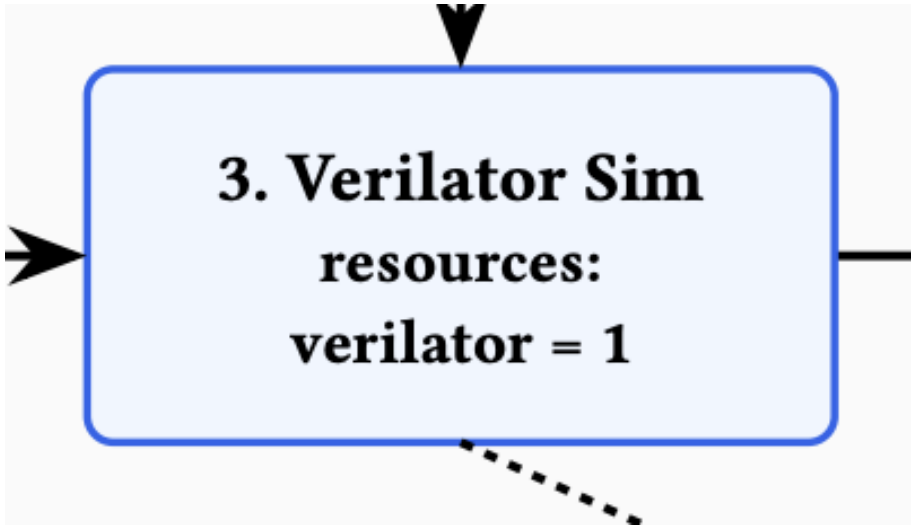
CHIA Abstractions: Workflows, Clusters, and Loops



CHIA Abstractions: Loops



CHIA Abstractions: Defining and Running Nodes

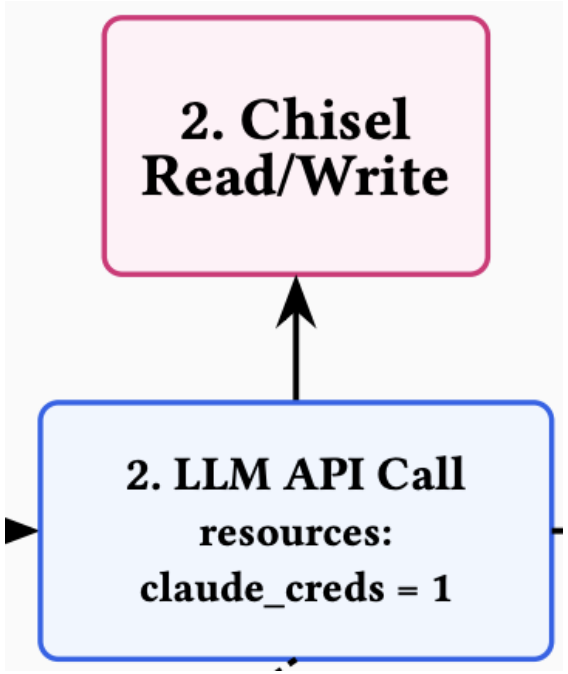


The Power of ChiaFunction:

1) Explicit Resource Demands
2) Simple Asynchronous/Parallel Execution
   1) On heterogeneous substrate
   2) At scale
3) Built-in Profiling
4) Automatic retry

```
@ChiaFunction(resources={"verilator": 1.0})
def verilate(sim_binary, test_binaries):
        .../


test_results_future =
        verilate.chia_remote(sim_bin, test_bins)
```

CHIA Abstractions: Agentic Orchestration



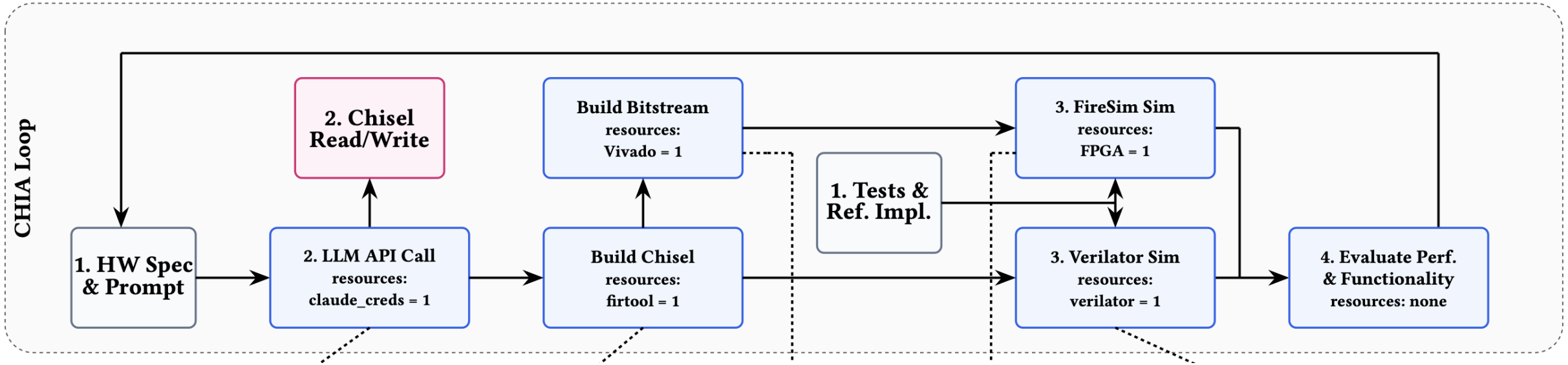
The Power of ChiaTool:

You can give the agents exactly as much control of the flow as you want to give them.

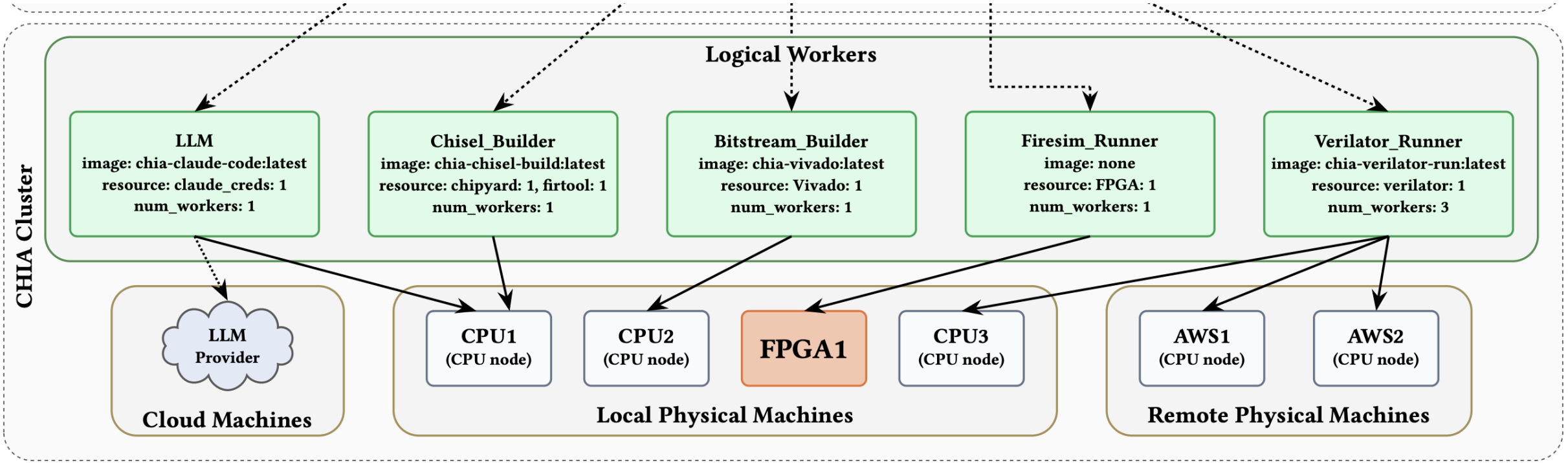
Let them orchestrate everything, or just let them reason, or anything in between.

```
@ChiaFunction()
def read_src(filename) -> str: …

@ChiaFunction()
def write_src(filename, contents): …

class rw_src_tool(ChiaTool):

def setup():
        self.mcp.add_tool(read_src,name="read_src_tool")
        self.mcp.add_tool(write_src,name="write_src_tool")
```

CHIA Abstractions: Loops



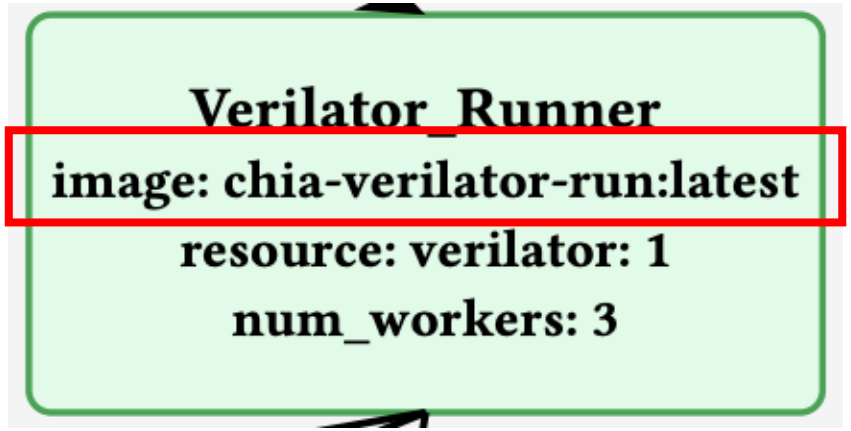
CHIA Abstractions: Clusters



The Power of CHIA:

Leverage on-premises and public cloud compute together in a single cluster.

CHIA Abstractions: Logical Workers




```
# Spec2RTL.yaml
available_node_types:
    Verilator_Runner:
        docker:
            image: "chia-verilator-run:latest"
        resources: {"verilator": 1}
        num_workers: 3
        compatible_ips: [CPU3 \
                         @verilator_aws:0, \
                         @verilator_aws:1]
```

The Power of CHIA:

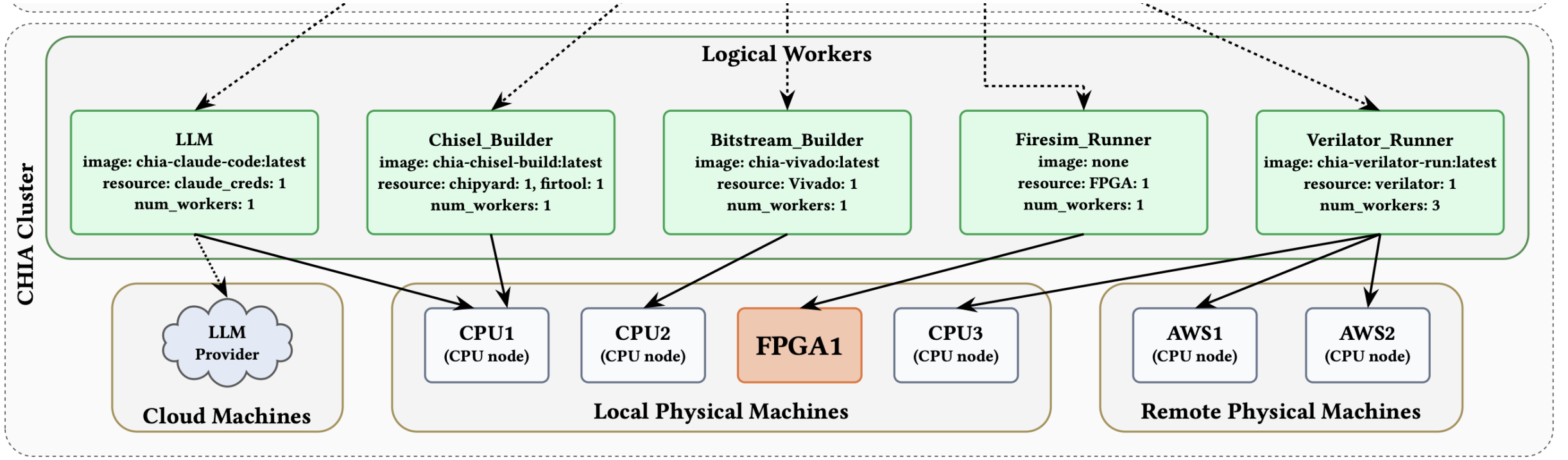
Logical Workers:

1. Clean SW resource abstraction
2. Abstracts heterogeneous HW resources

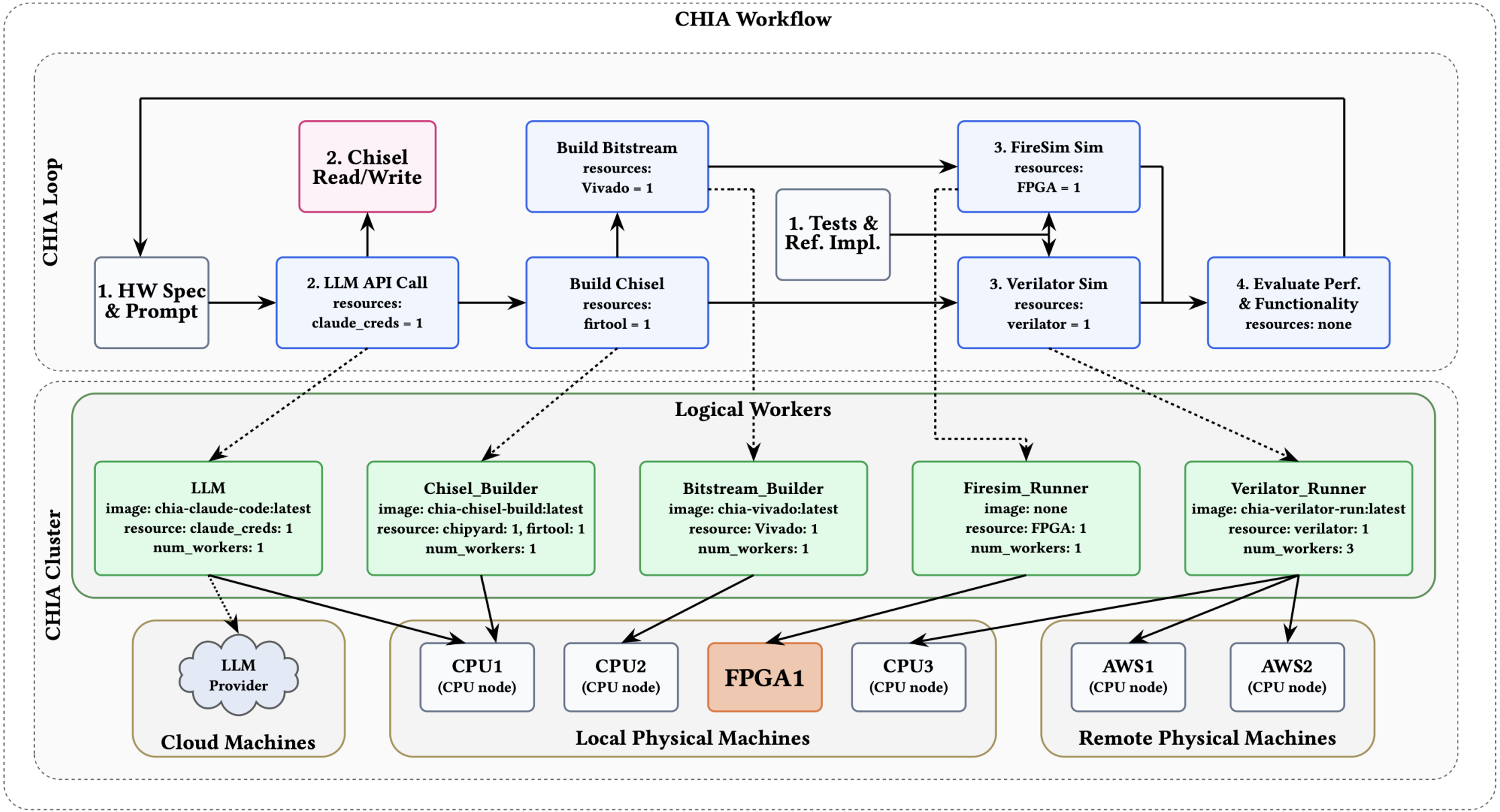
Containerization:

1. No dependency management hell
2. Isolates LLMs
3. Makes *Logical Workers* (more) portable

CHIA Abstractions: Clusters



CHIA Abstractions: Workflows, Clusters, and Loops



CHIA Library: CHIA works with the tools you already use!

Don’t see your tool here? Consider contributing a node to CHIA!

CHIPYARD

gem5

ChampSim

FireSim

VERILATOR

HAMMER

(commercial CAD tools)

SkyDiscover

AlphaEvolve

OpenEvolve

Claude Code

Antigravity

Codex

Case Studies

Case Studies Overview (see paper for details!)

1

Agentic generation of a gem5 model from BOOM RTL

Improve gem5 core model's accuracy from 40% → 97.2% relative to ground-truth RTL simulation of BOOM over 10.5 days on 36 benchmarks

2

Agentic RISC-V ISA Extension RTL Implementation

5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change

3

Agentic IPC-aware Critical-Path Optimization in RTL

2.03× frequency increase with only 3.28% IPC loss on SPEC06 ref. by rewriting RTL; net 1.96× Iron Law speedup; negligible area change

4

SkyDiscover

AlphaEvolve

Agentic Architecture Discovery in µArch Simulators and RTL

AI agent and evolutionary algorithm-directed exploration and implementation of novel microarchitectural ideas in ChampSim & BOOM

5

Maintainer-friendly Agentic Bug-fixing for CIRCT GitHub Issues

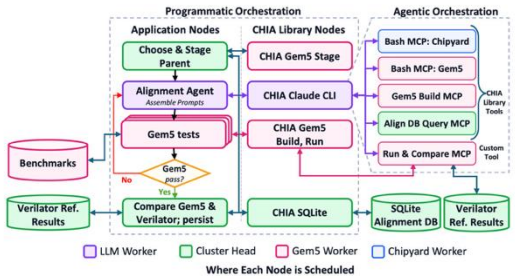
Autonomously triaged CIRCT GitHub issues into actionable, high-quality, and now upstreamed PRs in ~45 minutes

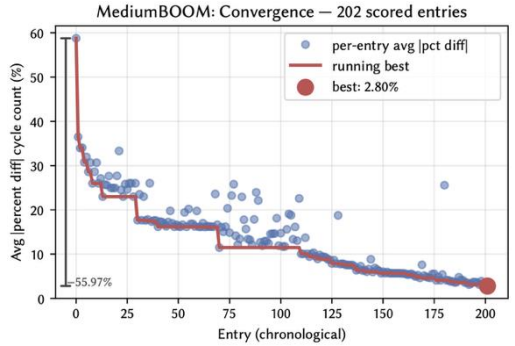
Case Studies Overview (see paper for details!)

1 gem5

Agentic generation of a gem5 model from BOOM RTL

Improve gem5 core model's accuracy from 40% → 97.2% relative to ground-truth RTL simulation of BOOM over 10.5 days on 36 benchmarks

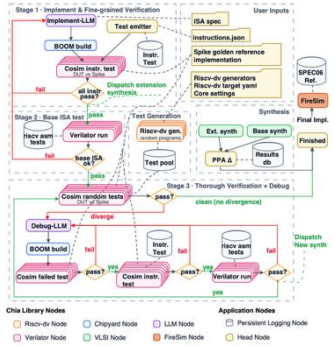


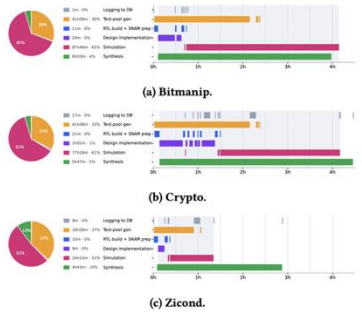


2 RISC-V

Agentic RISC-V ISA Extension RTL Implementation

5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change

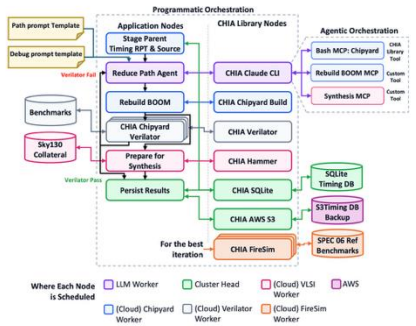




3

Agentic IPC-aware Critical-Path Optimization in RTL

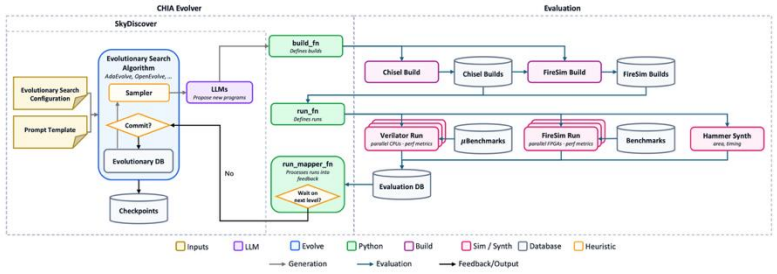
2.03× frequency increase with only 3.28% IPC loss on SPEC06 ref. by rewriting RTL; net 1.96× Iron Law speedup; negligible area change



4 SkyDiscover / AlphaEvolve

Agentic Architecture Discovery in µArch Simulators and RTL

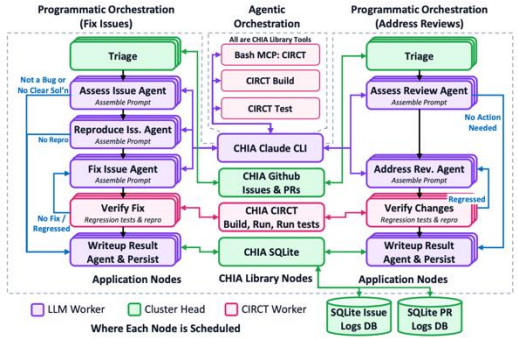
AI agent and evolutionary algorithm-directed exploration and implementation of novel microarchitectural ideas in ChampSim & BOOM



5 GitHub / CIRCT

Maintainer-friendly Agentic Bug-fixing for CIRCT GitHub Issues

Autonomously triaged CIRCT GitHub issues into actionable, high-quality, and now upstreamed PRs in ~45 minutes

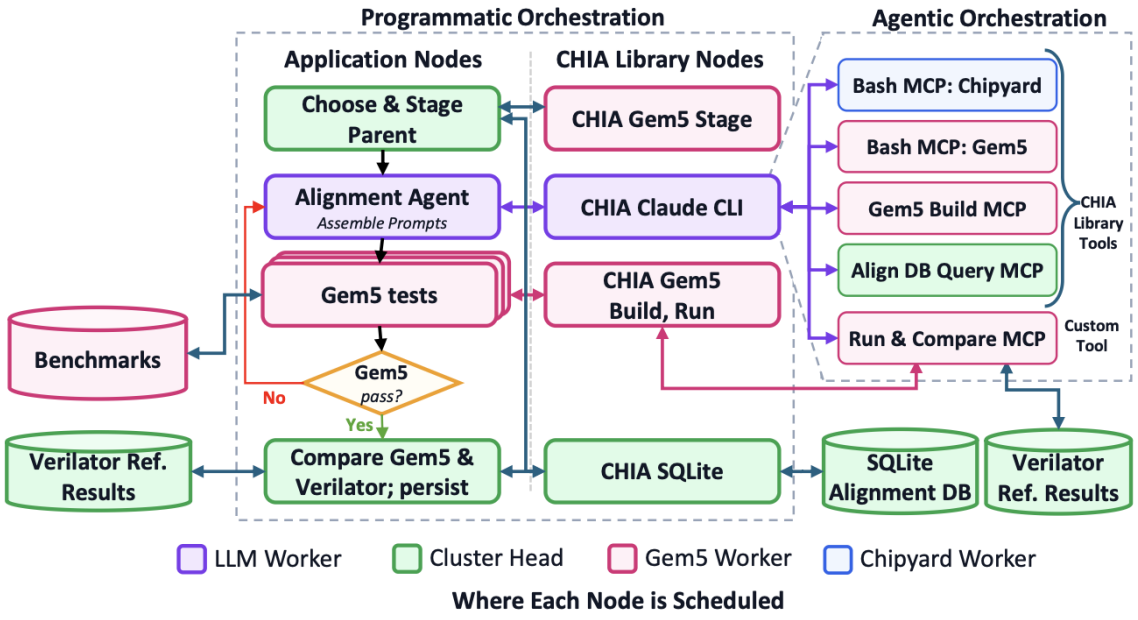


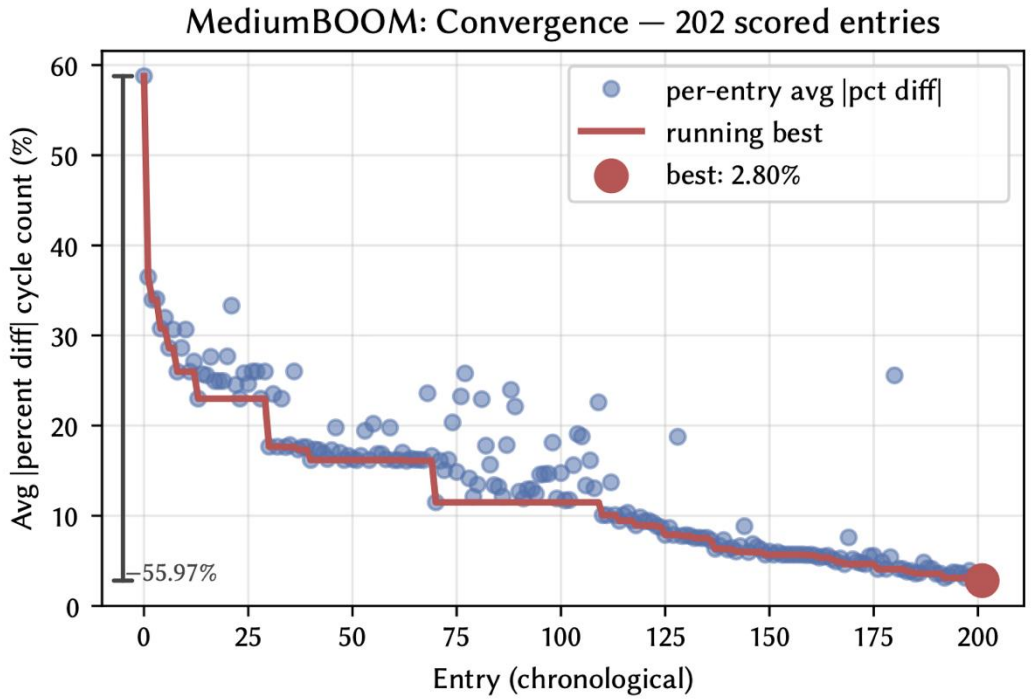
Case Studies Overview (see paper for details!)

1

Agentic generation of a gem5 model from BOOM RTL

Improve gem5 core model's accuracy from 40% → 97.2% relative to ground-truth RTL simulation of BOOM over 10.5 days on 36 benchmarks





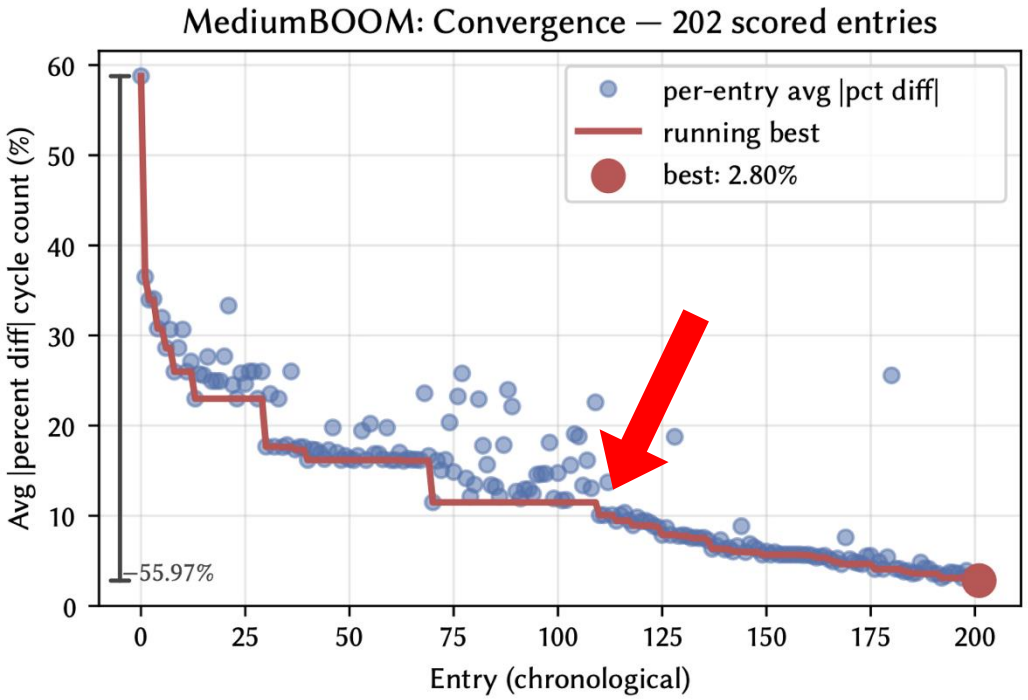
Case Studies Overview (see paper for details!)

1

Agentic generation of a gem5 model from BOOM RTL

Improve gem5 core model's accuracy from 40% → 97.2% relative to ground-truth RTL simulation of BOOM over 10.5 days on 36 benchmarks

1. Flexibility and experimentation
2. Many loop designs tried in parallel --> easy to experiment
3. 1h to build!
4. Iteration 110:
   The LLM can now test its own changes

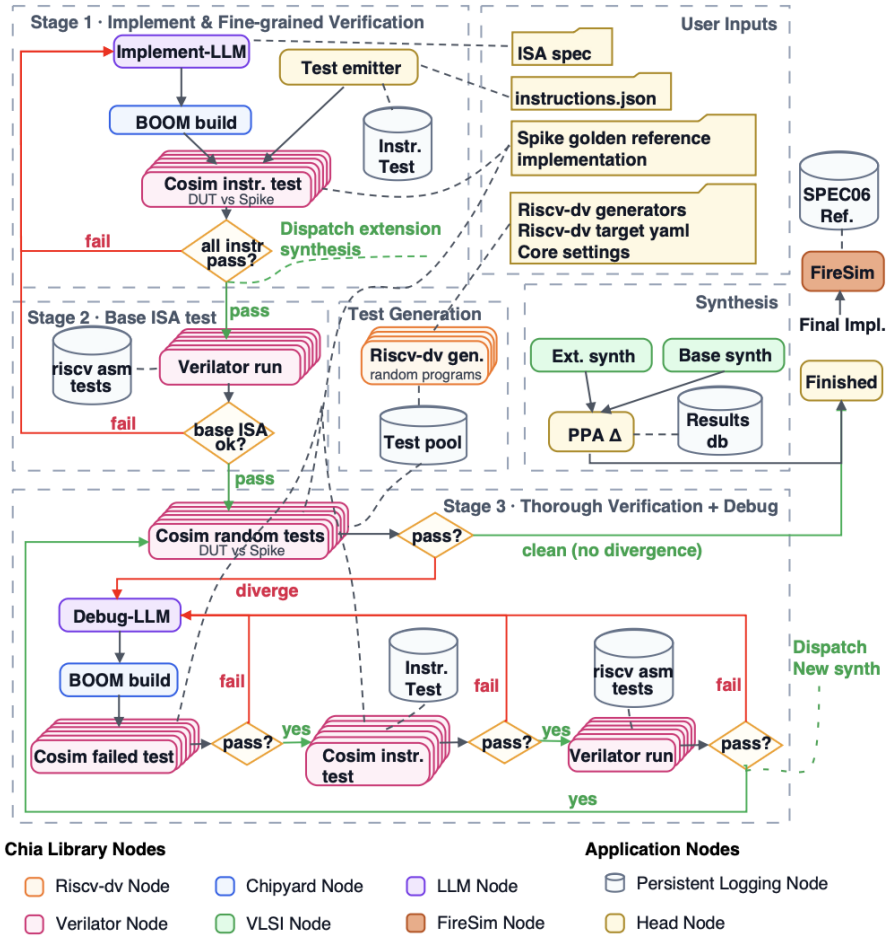


Case Studies Overview (see paper for details!)

2

Agentic RISC-V ISA Extension RTL Implementation

5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change

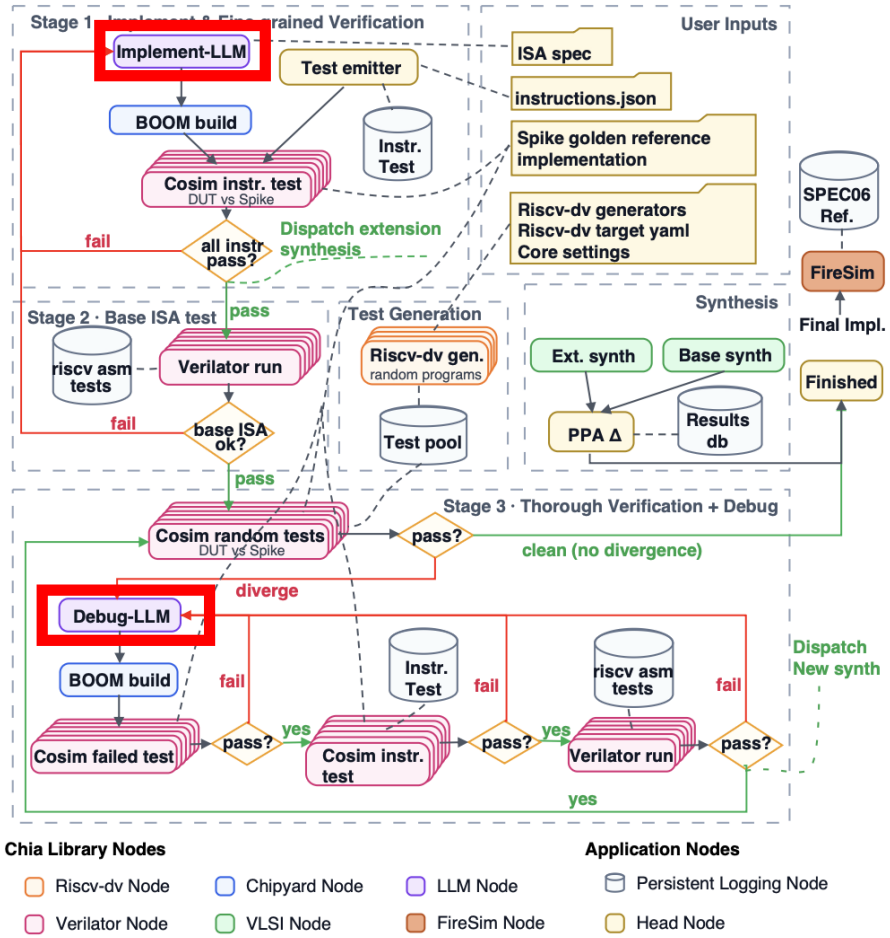


Case Studies Overview (see paper for details!)

2

Agentic RISC-V ISA Extension RTL Implementation

5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change



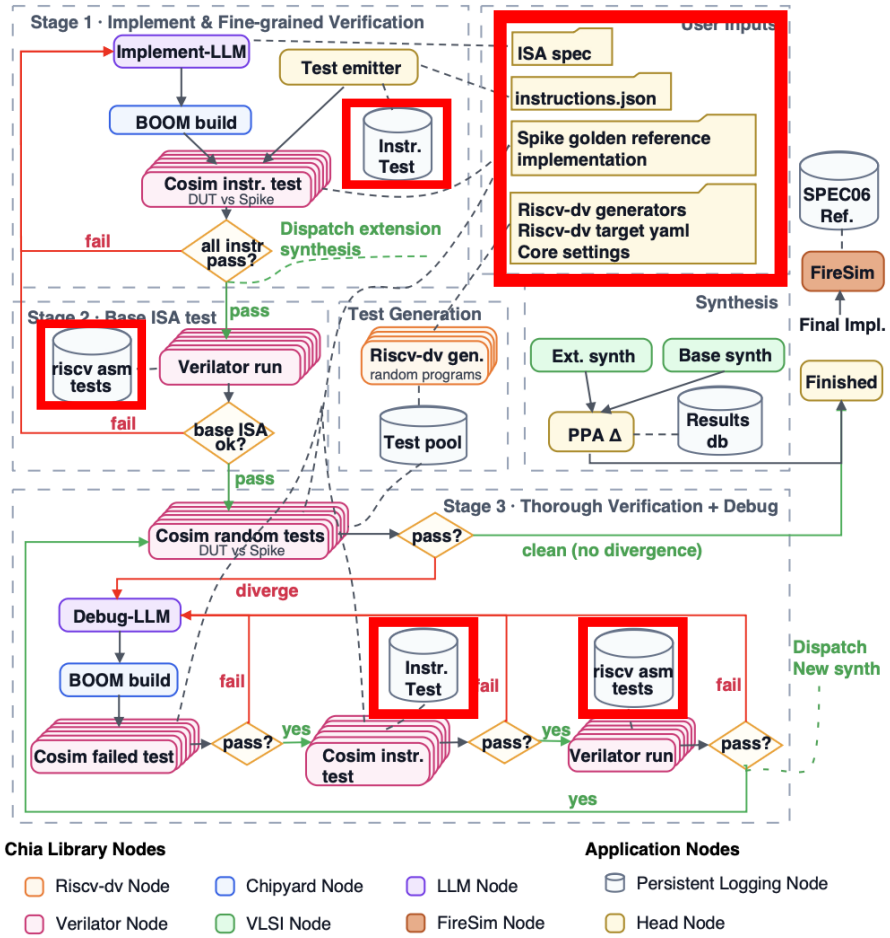
1. CHIA + LLMs for specific tasks

Case Studies Overview (see paper for details!)

2

Agentic RISC-V ISA Extension RTL Implementation

5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change



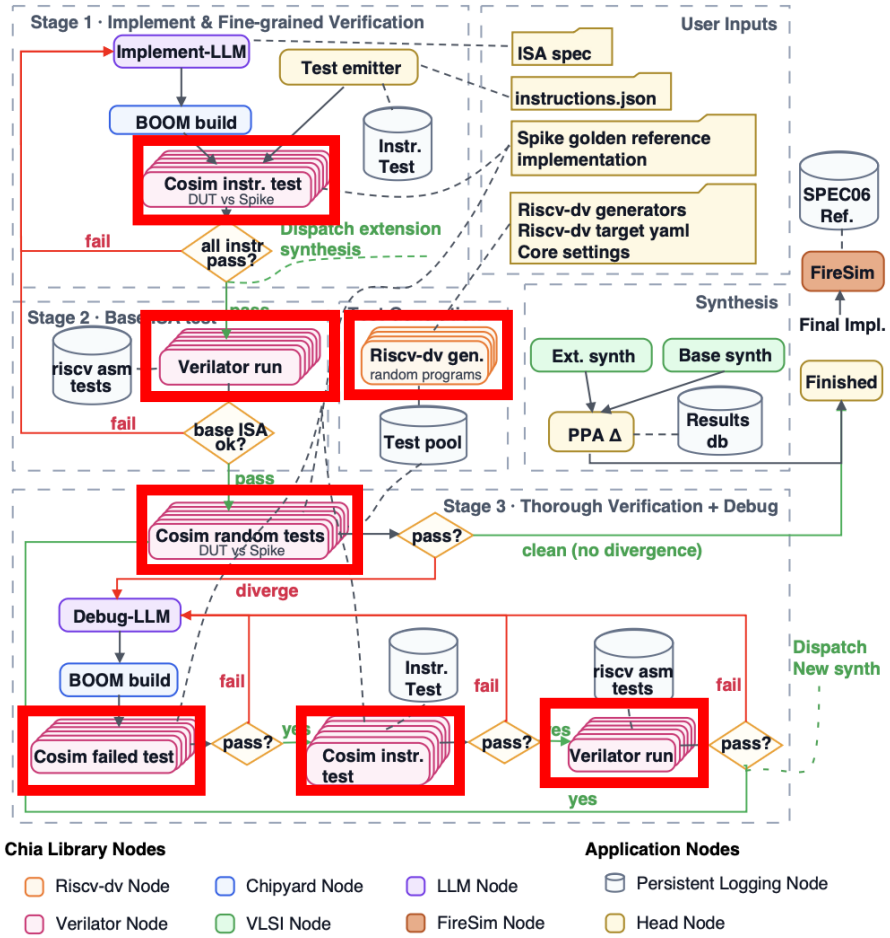
1. CHIA + LLMs for specific tasks
2. Control where it matters

Case Studies Overview (see paper for details!)

2

Agentic RISC-V ISA Extension RTL Implementation

5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change



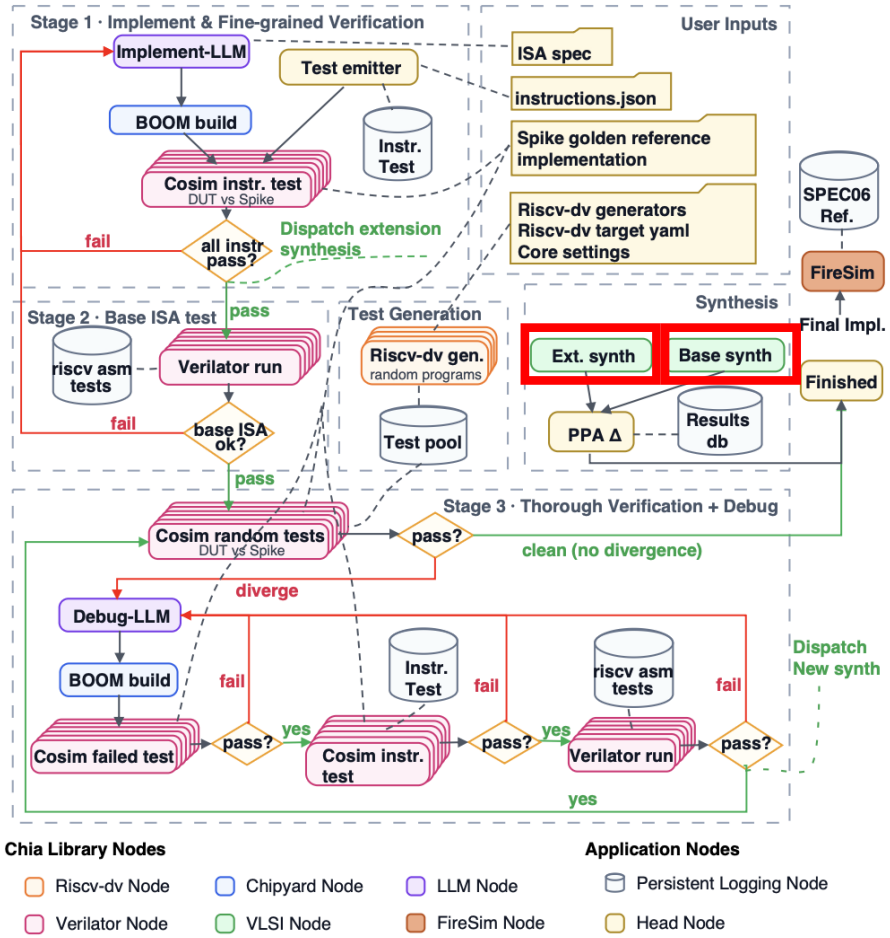
1. CHIA + LLMs for specific tasks
2. Control where it matters
3. Resource management

Case Studies Overview (see paper for details!)

2

Agentic RISC-V ISA Extension RTL Implementation

5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change



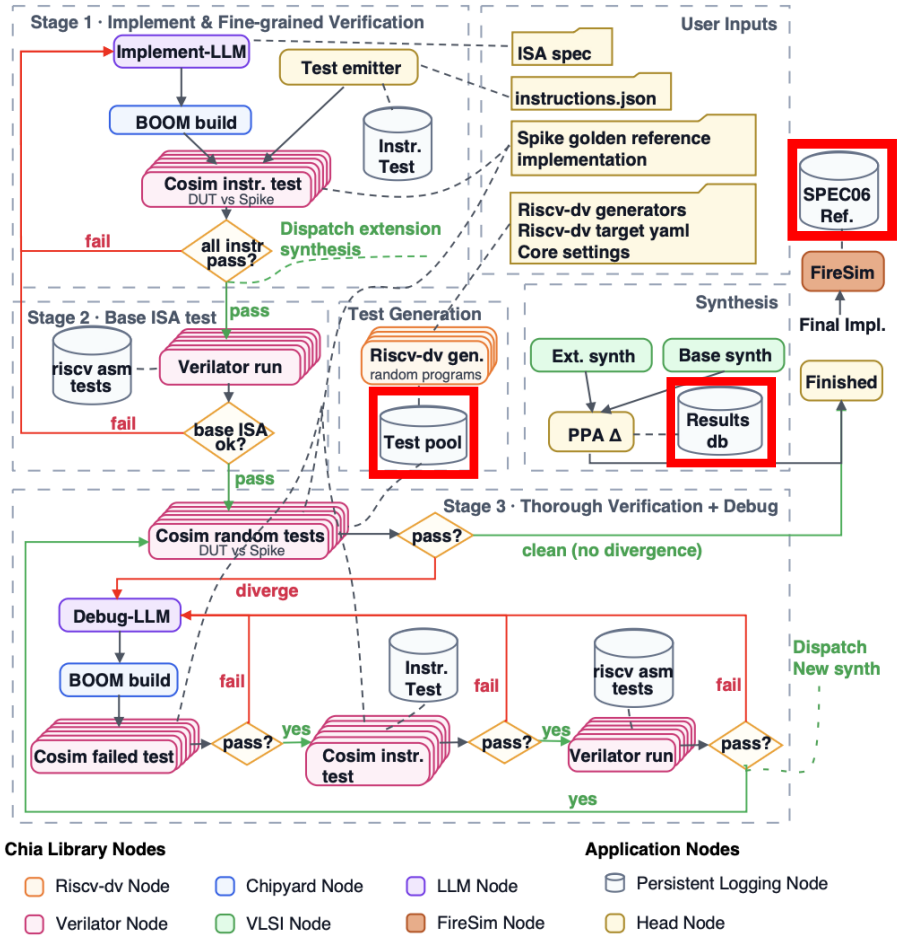
1. CHIA + LLMs for specific tasks
2. Control where it matters
3. Resource management
4. Licenses stay on-prem

Case Studies Overview (see paper for details!)

2

Agentic RISC-V ISA Extension RTL Implementation

5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change



1. CHIA + LLMs for specific tasks
2. Control where it matters
3. Resource management
4. Licenses stay on-prem
5. Data logging

Case Studies Overview (see paper for details!)

2

Agentic RISC-V ISA Extension RTL Implementation

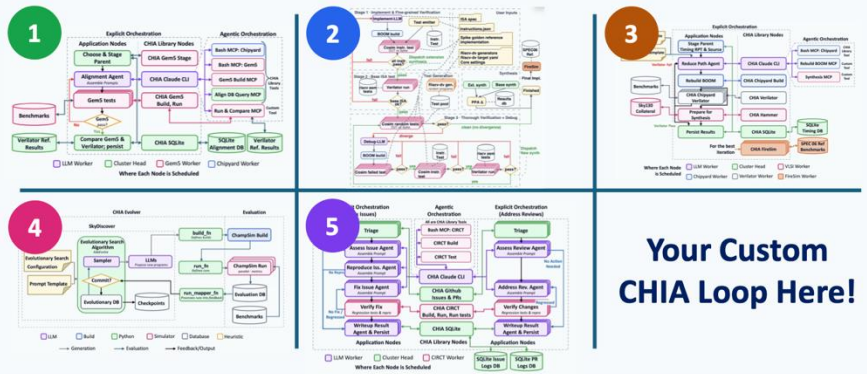
5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change

In less than 5h…

Bitmanip	Zicond	Crypto
5.6% on SPEC Ref.	3.5% on SPEC Ref.	10x OpenSSL Speed Crypto
Entire 25+ Trillion instr.	Entire 25+ Trillion instr.	Entire 200B instructions
2% area increase, sky130	no area increase, sky130	5% area increase, sky130
No max freq. regression	No max freq. regression	No max freq. regression

Summary

CHIA Loops



CHIA

CHIA Cluster

Hybrid On-Premises + Cloud Deployments

CPU GPU FPGA API Storage

Case Studies

1 **Agentic generation of a gem5 model from BOOM RTL**
Improve gem5 core model's accuracy from 40% → 97.2% relative to ground-truth RTL simulation of BOOM over 10.5 days on 36 benchmarks

2 **Agentic RISC-V ISA Extension RTL Implementation**
5.6% (Bitmanip) and 3.5% (Zicond) speedup on SPEC06 ref., 25+ T insts.; 10× speedup on OpenSSL (Crypto); < 5% area impact & no freq. change

3 **Agentic IPC-aware Critical-Path Optimization in RTL**
2.03× frequency increase with only 3.28% IPC loss on SPEC06 ref. by rewriting RTL; net 1.96× Iron Law speedup; negligible area change

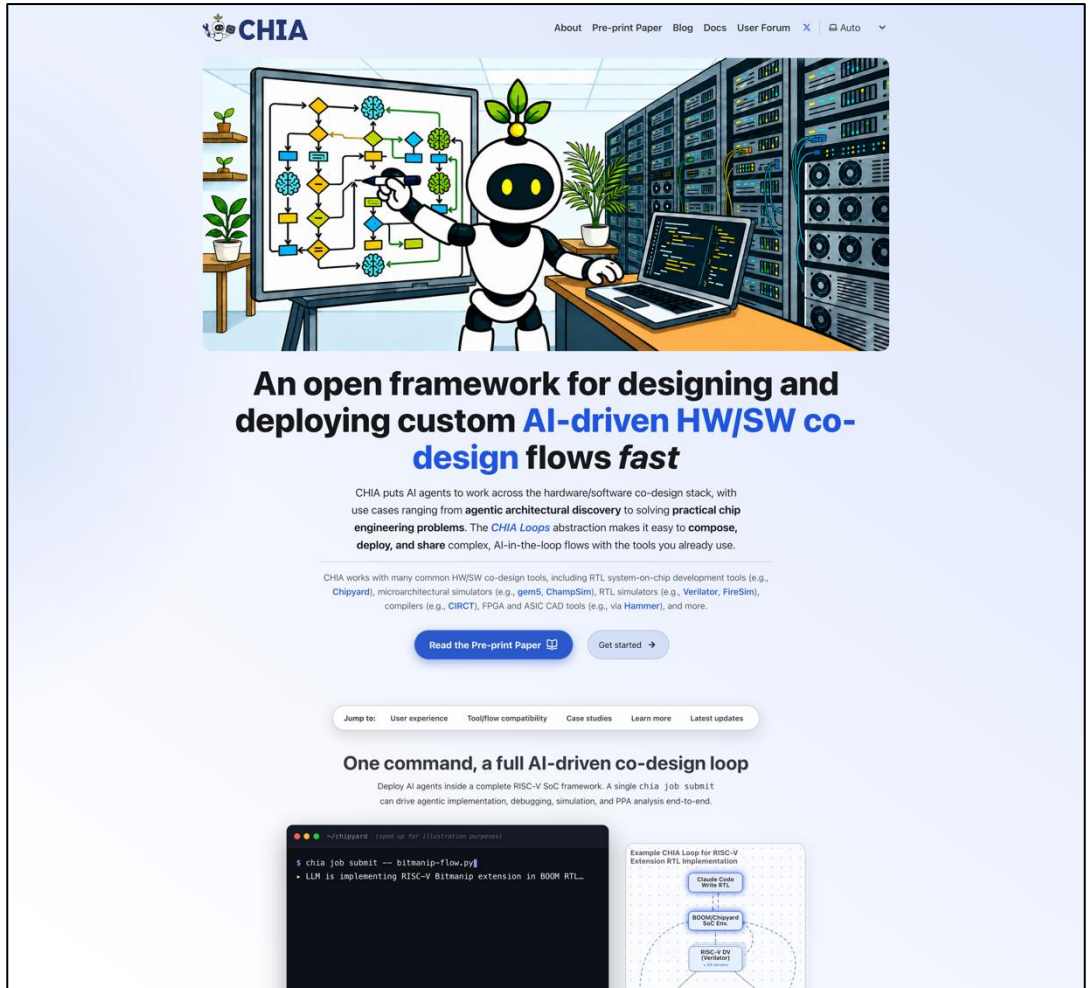
4 **Agentic Architecture Discovery in µArch Simulators and RTL**
AI agent and evolutionary algorithm-directed exploration and implementation of novel microarchitectural ideas in ChampSim & BOOM

5 **Maintainer-friendly Agentic Bug-fixing for CIRCT GitHub Issues**
Autonomously triaged CIRCT GitHub issues into actionable, high-quality, and now upstreamed PRs in ~45 minutes

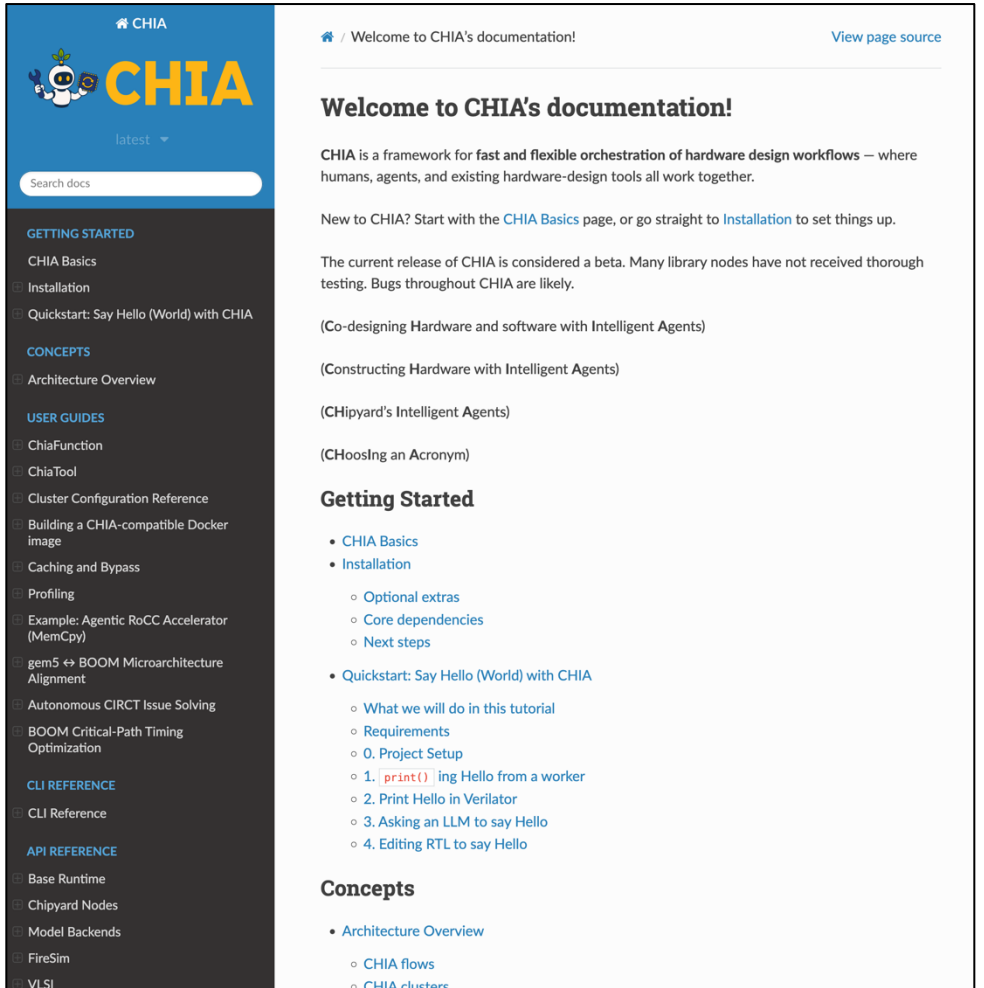
Your Custom CHIA Loop Here!
Build your own custom AI-driven HW/SW co-design loops with CHIA!

CHIA is hot off the presses, available to use!

https://chialoops.ai



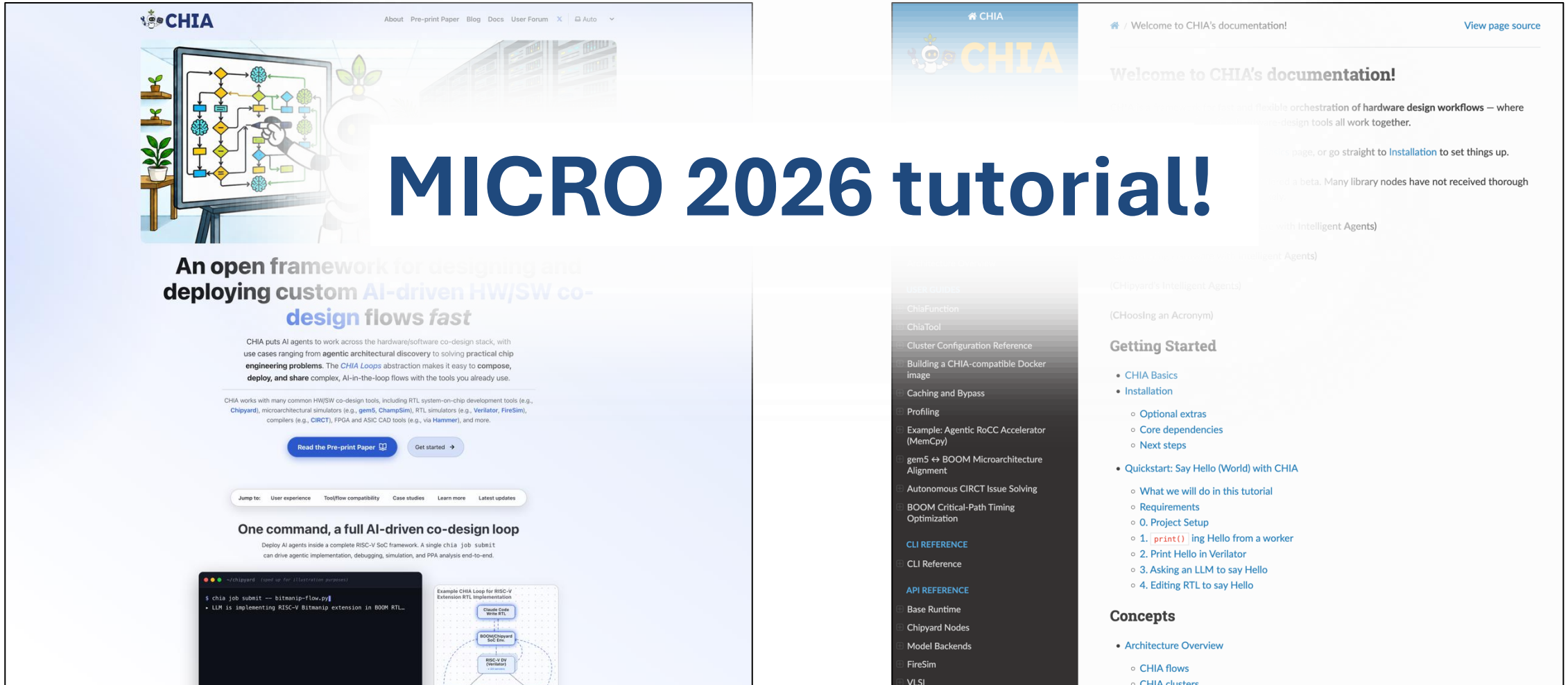
https://docs.chialoops.ai



CHIA is hot off the presses, available to use!

https://chialoops.ai

https://docs.chialoops.ai



MICRO 2026 tutorial!

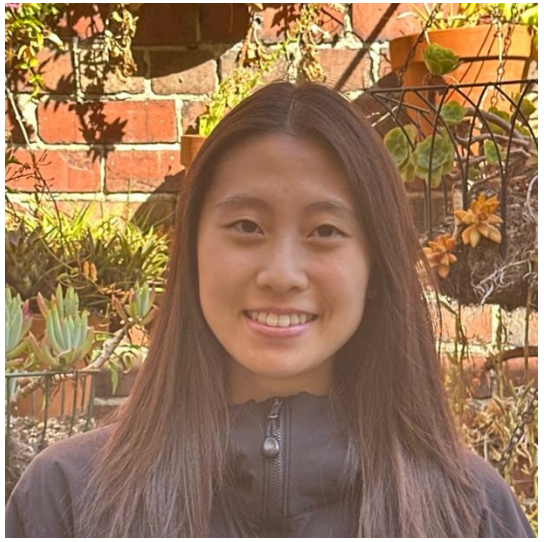
CHIA

https://chialoops.ai

(includes paper link)

Come say hi!

(P.S. We have stickers)



Angela Cui

Ferran Hermida-Rivera

Jack Toubes

Berkeley Architecture Research