

# ASTRA-sim and Chakra Tutorial:

## *Overview of Chakra and ASTRA-sim*

Tushar Krishna

Associate Professor

School of ECE, Georgia Institute of Technology

[tushar@ece.gatech.edu](mailto:tushar@ece.gatech.edu)



# ASTRA-sim Tutorial - Agenda

Time (PDT)	Topic	Presenter
3:00 – 3:30 pm	Introduction to Distributed ML	Tushar Krishna
3:30 – 3:45 pm	<b>Overview of Chakra and ASTRA-sim</b>	<b>Tushar Krishna</b>
3:45 – 4:35 pm	<b>Deeper Dive into Chakra and ASTRA-sim</b>	Will Won
	Workload, System, and Network Layers	
4:35 – 4:45 pm	<b>Demo</b>	Will Won
4:45 – 5:00 pm	<b>Closing Remarks</b>	Tushar Krishna

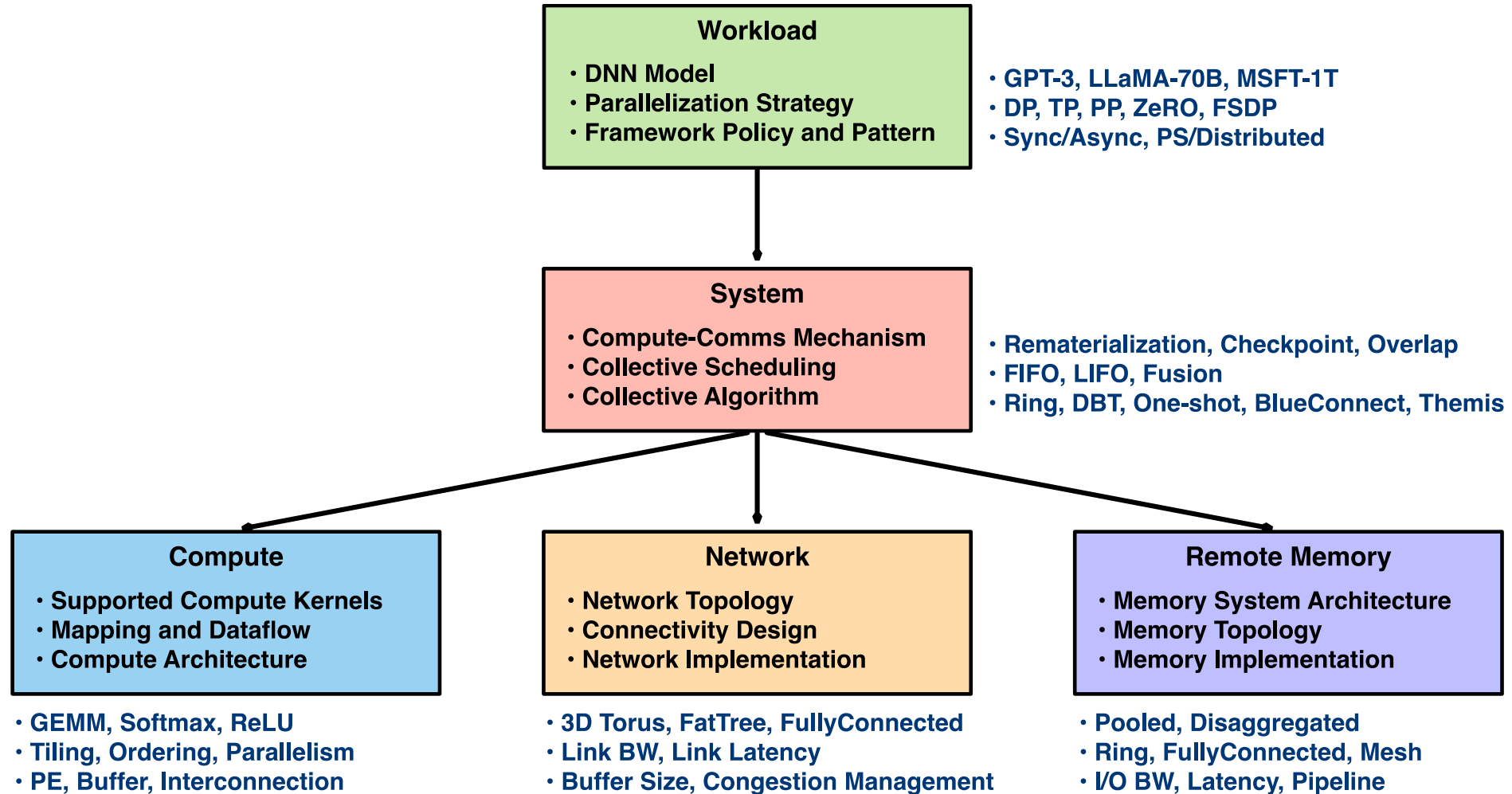
## Tutorial Website

*includes agenda, slides, ASTRA-sim installation instructions (via source + docker image)*

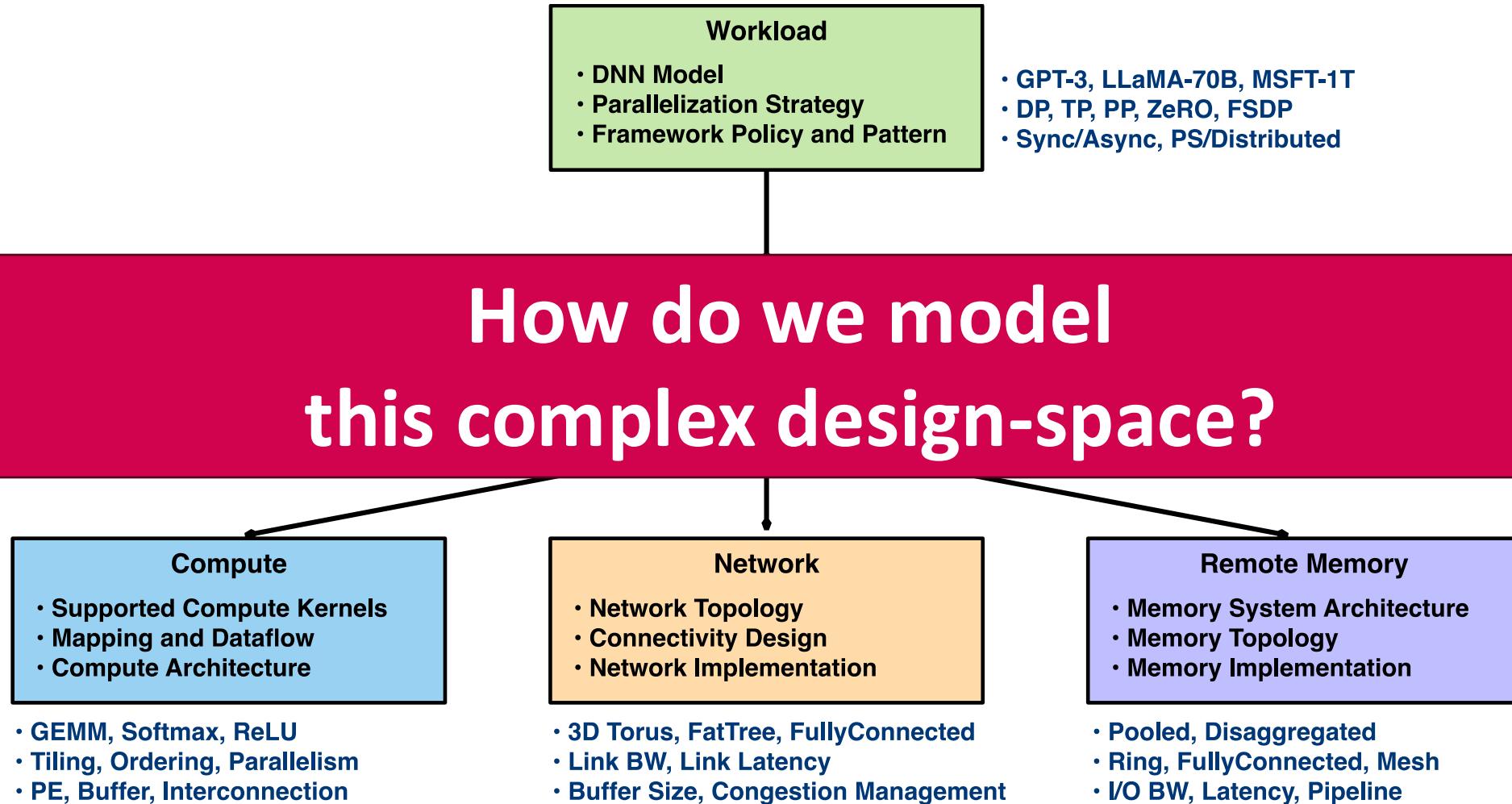
**<https://astra-sim.github.io/tutorials/hoti-2024>**

**Attention:** Tutorial is being recorded

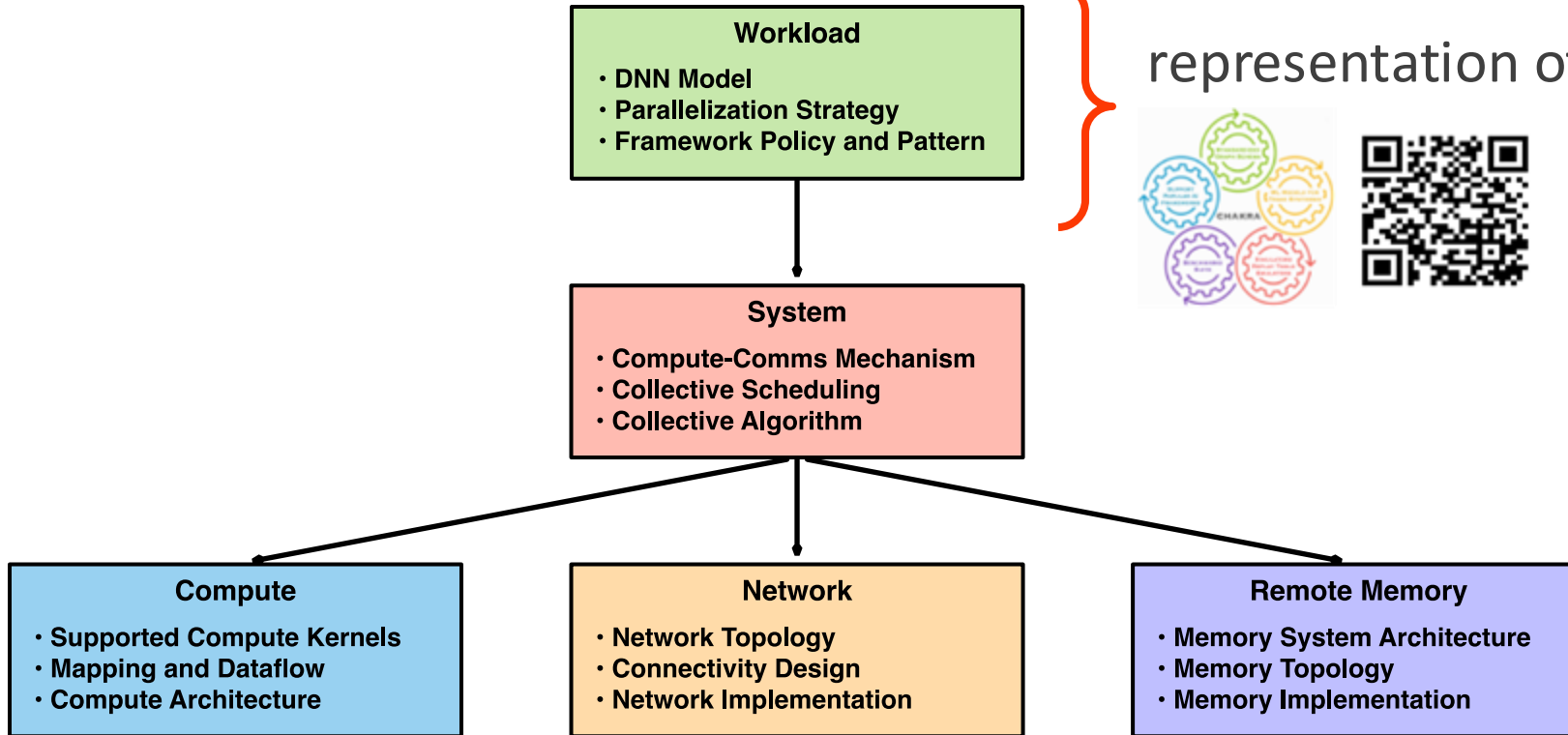
# Challenge: Complex SW/HW Co-Design Space



# Challenge: Complex SW/HW Co-Design Space



# Introducing Chakra and ASTRA-sim



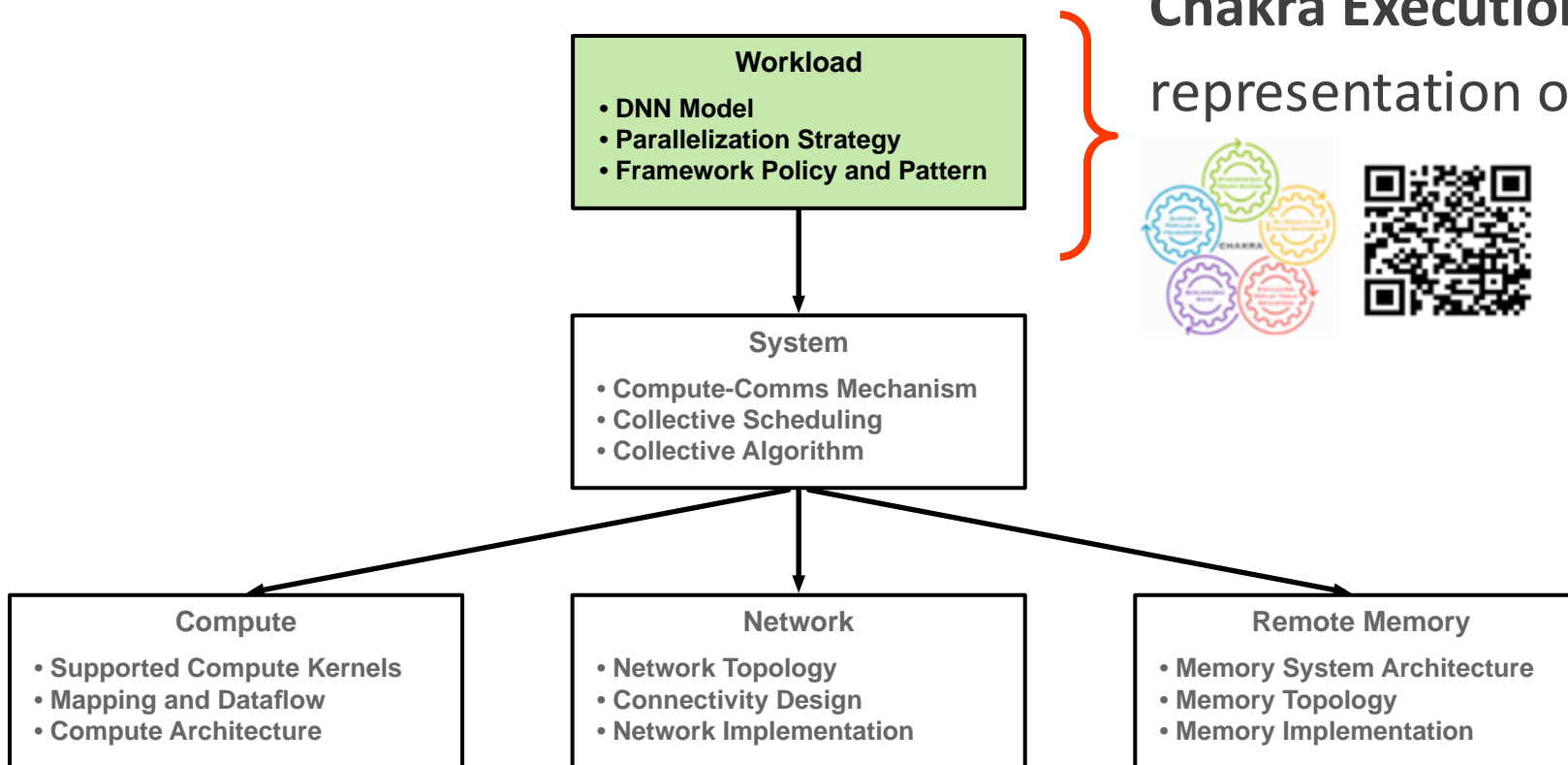
**Chakra Execution Trace:** an open graph-based representation of AI/ML workload execution



**ASTRA-sim:**  
Distributed AI system simulator



# Introducing Chakra and ASTRA-sim



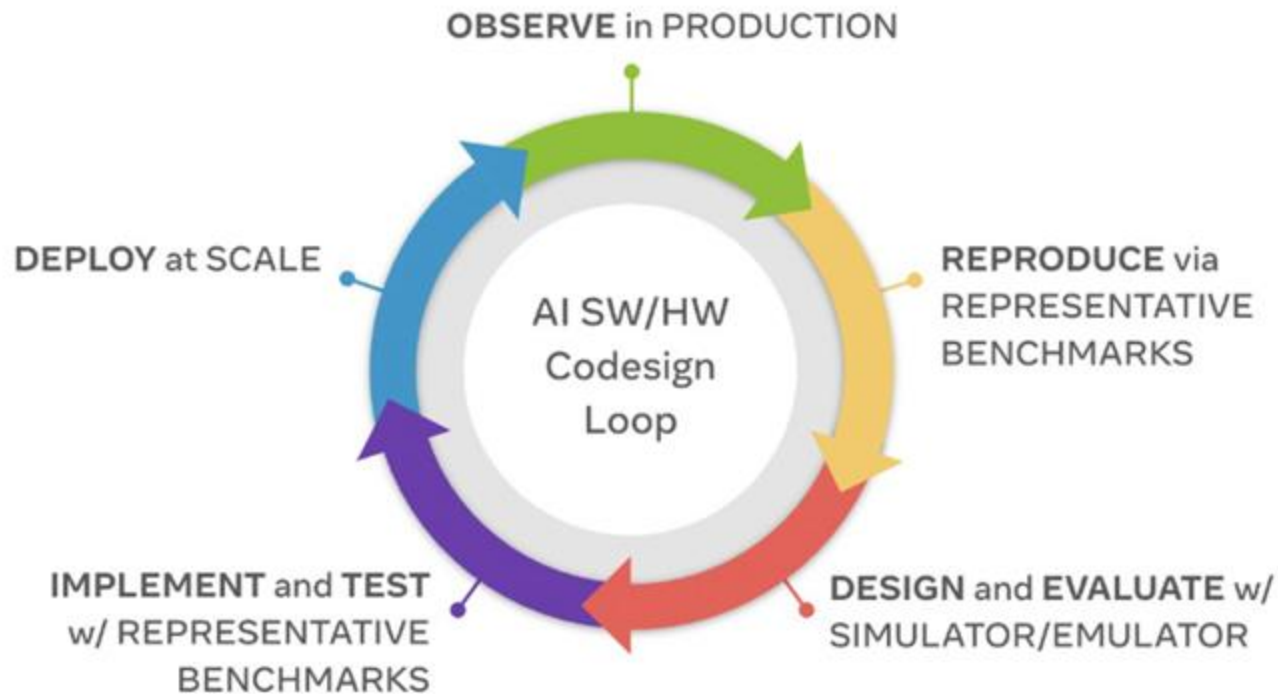
**Chakra Execution Trace:** an open graph-based representation of AI/ML workload execution



**ASTRA-sim:**  
Distributed AI system simulator



# Chakra: Motivation

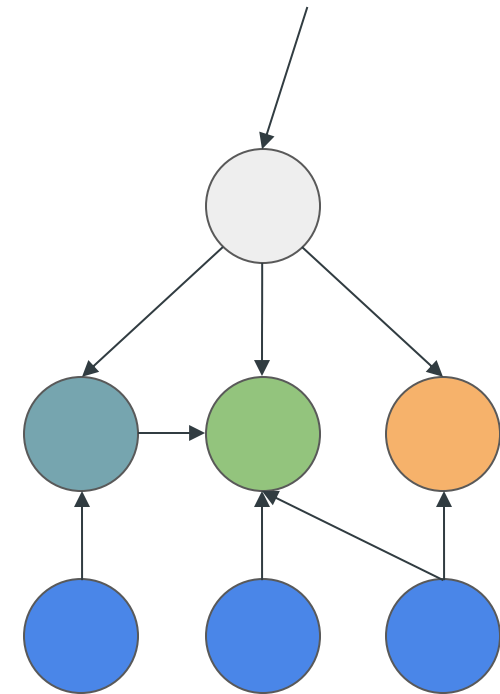


## Motivation

- High-cost of running full workload benchmarks
- Requires cross-domain full-stack expertise
- Difficult to isolate specific HW/SW bottlenecks
- Difficult to isolate compute, memory, network behavior
- Cannot keep up with the pace of AI innovation
- Hard to obfuscate proprietary AI model details
- Hard to reproduce without support infrastructure

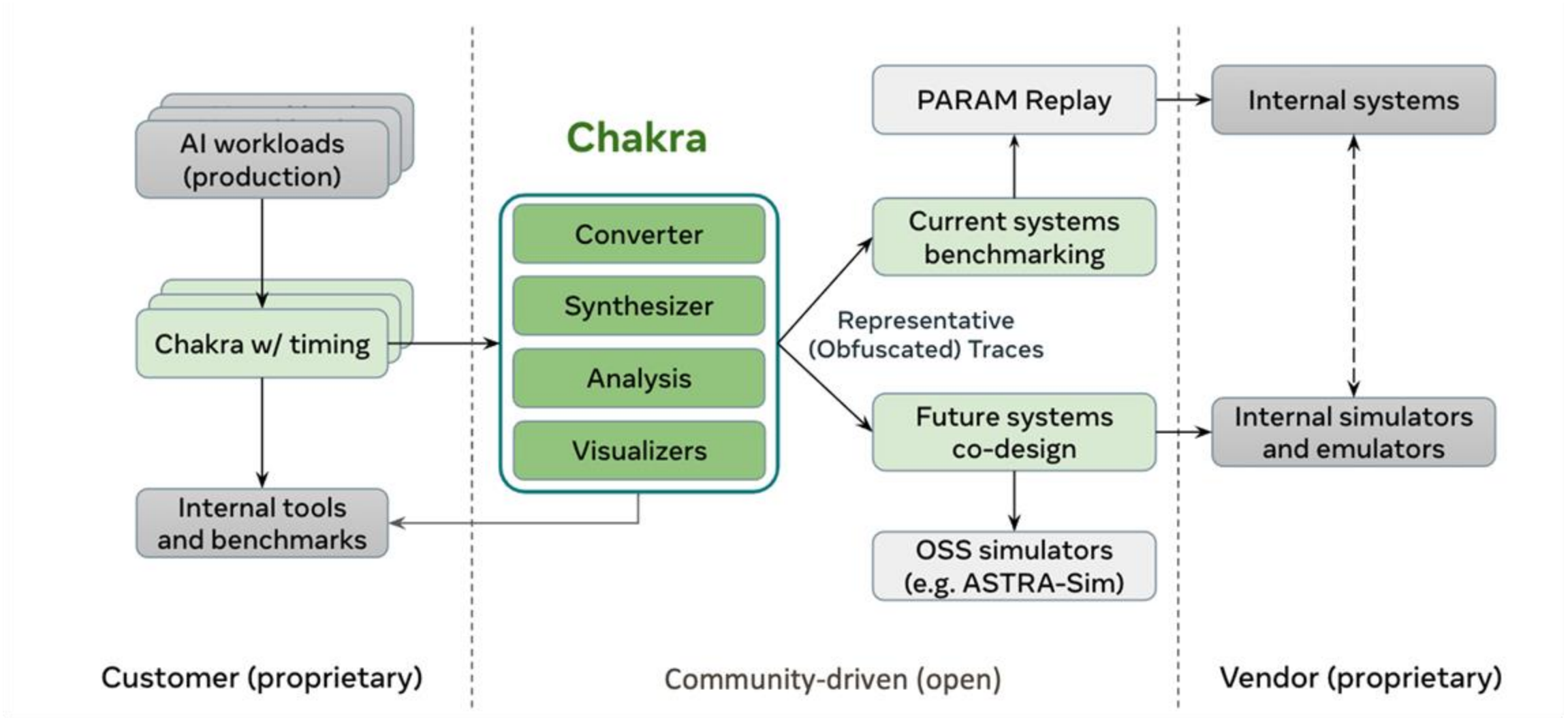
# Chakra Execution Traces

- **Hierarchical DAG**
- **Nodes**
  - Primitive operators: compute, comms, memory
  - Tensor objects: shape, size, device (local/remote)
  - Timing and resource constraints
- **Edges**
  - Data dependency
  - Control dependency (e.g. call stack)
- **Higher-level abstractions (e.g., components)**
  - Comprises of other components or primitive ops





# Chakra Ecosystem and End-to-End Flow

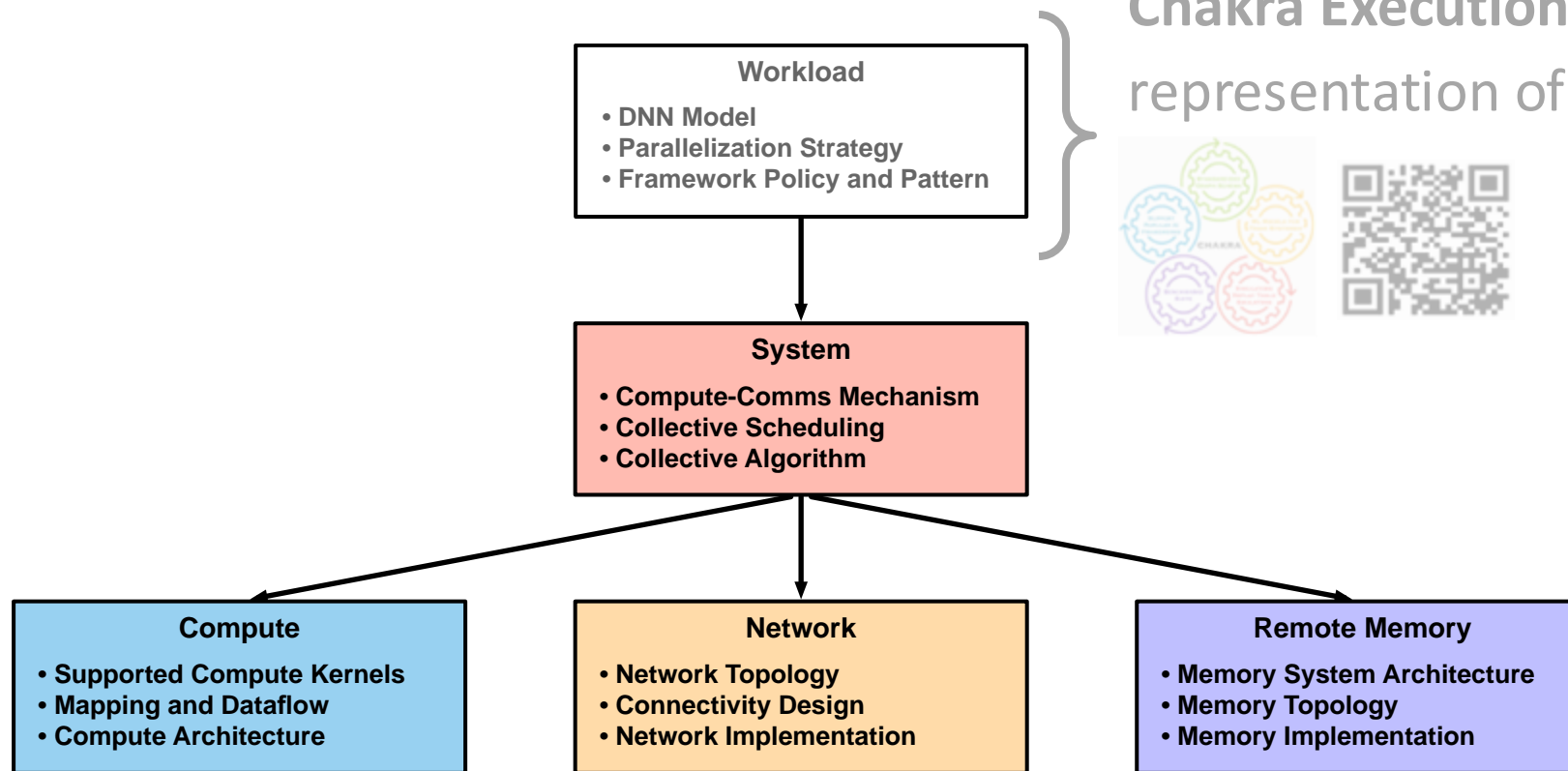


# Chakra is now part of MLCommons!



- **Build consensus on Execution Trace methodology**
  - Enable easier sharing between hyperscaler/cloud and vendors (with/without NDA)
  - Vendors can focus on different components (compute/memory/network)
  - Enable faster ramp-up for startups and academia
- **Shared engineering effort towards open/vibrant ecosystem**
  - Trace collection and synthesis
  - Support tools and downstream enablement
- **Benchmark suite definition and supervision**
  - Single workload and datacenter-scale benchmark scoring
  - Future workload projection

# Introducing Chakra and ASTRA-sim



**Chakra Execution Trace:** an open graph-based representation of AI/ML workload execution



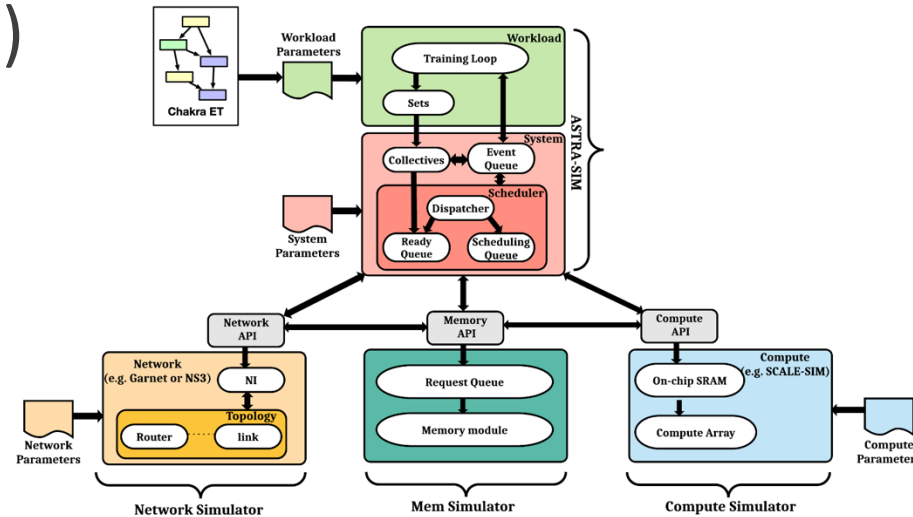
**ASTRA-sim:**  
Distributed AI system simulator



# ASTRA-sim: Design Principles

A **framework** to model/simulate/emulate AI systems with varying degrees of fidelity.

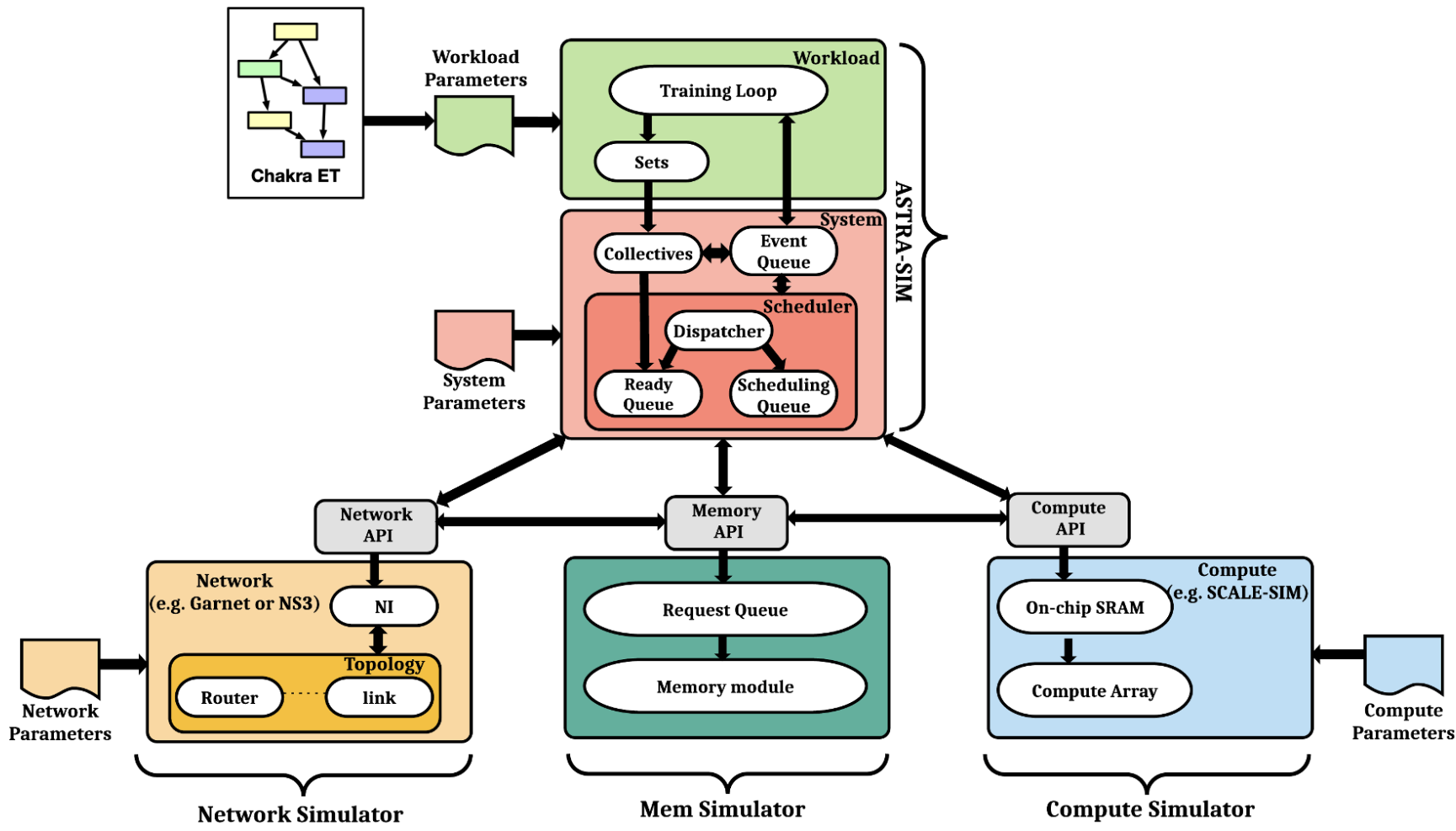
**Key enabler:** APIs for plugging in diverse external tools (i.e., composable simulators)



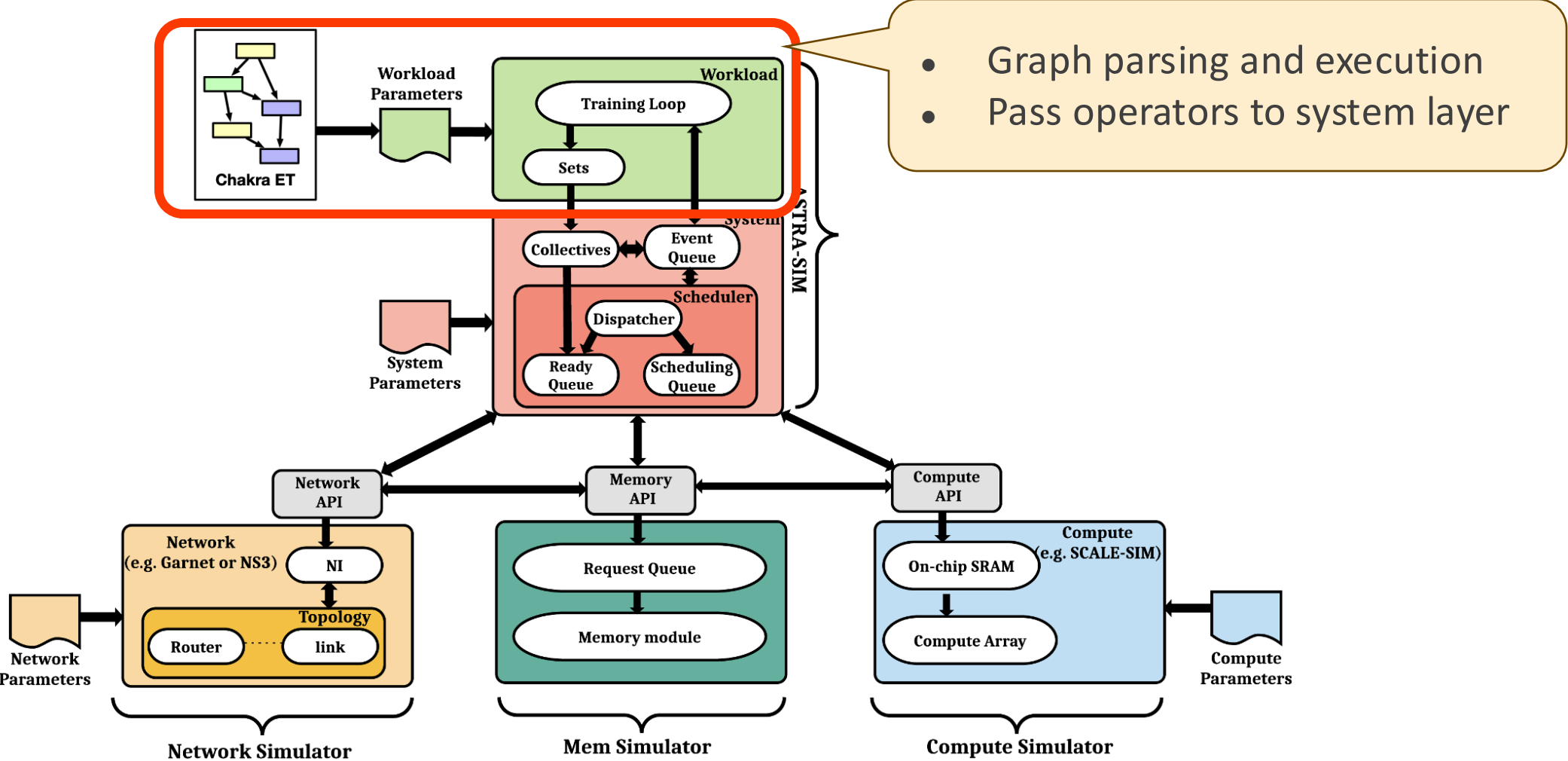
Reference Implementation: <http://github.com/astra-sim/astra-sim>

Website: <https://astra-sim.github.io/>

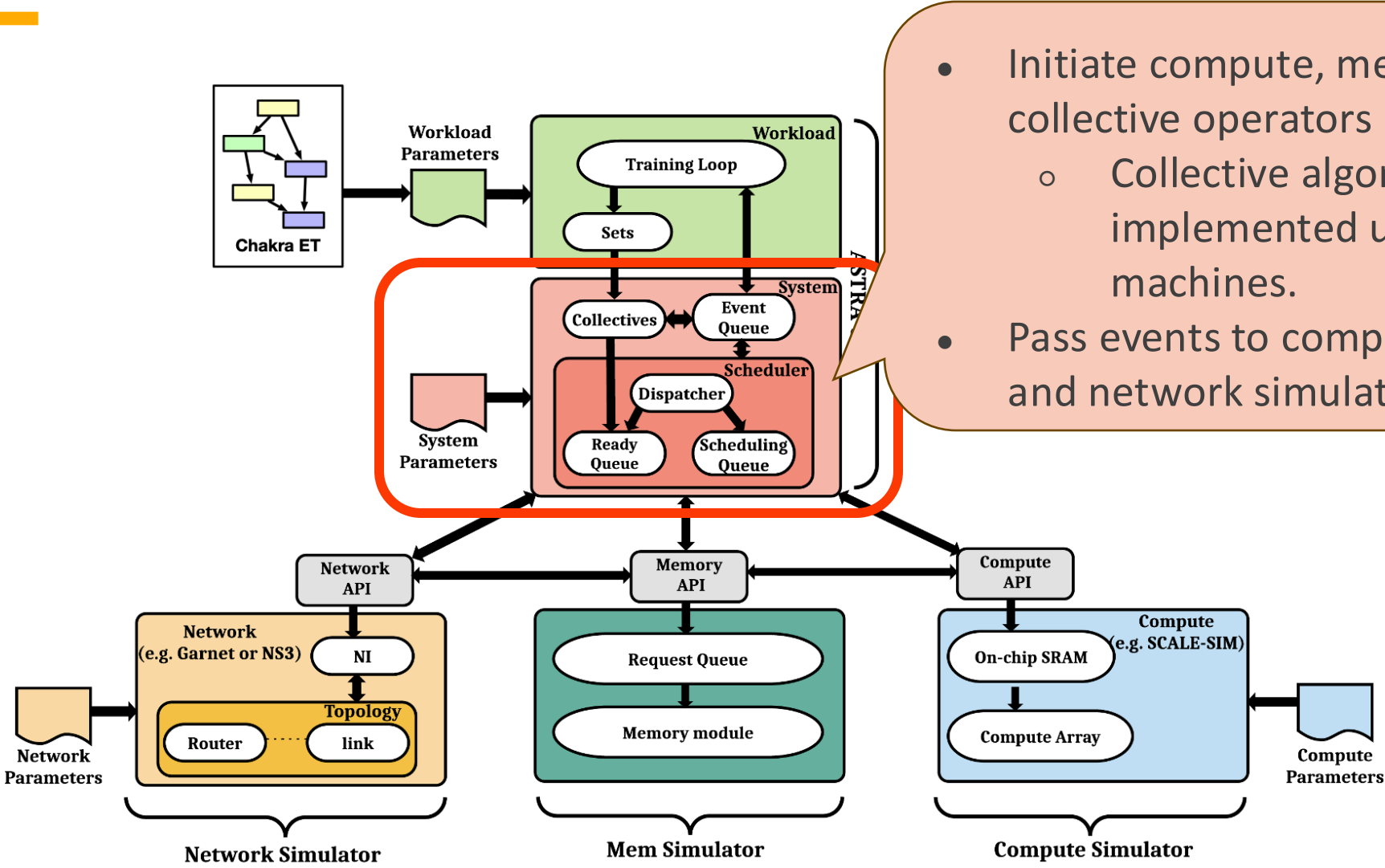
# ASTRA-sim



# ASTRA-sim: Workload Layer

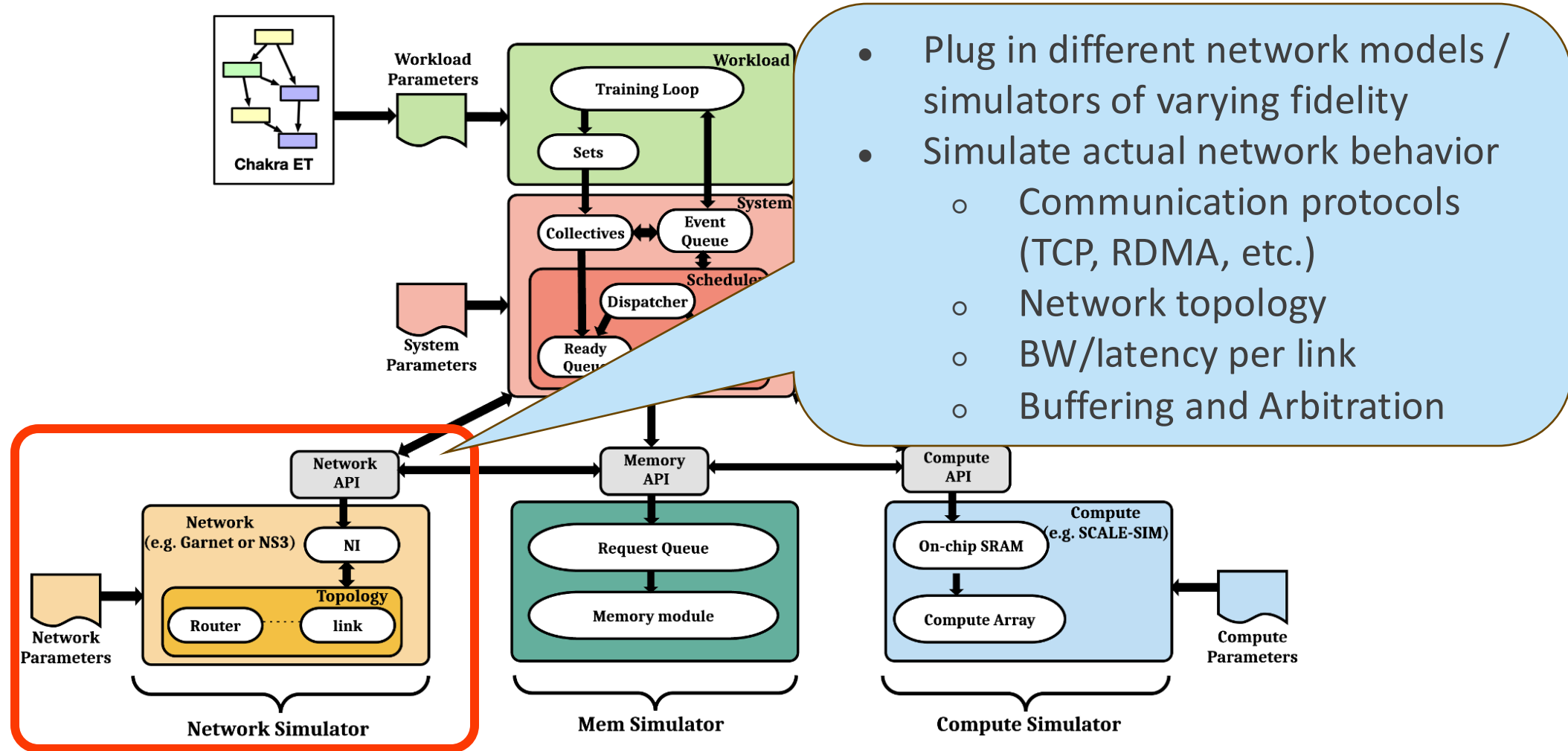


# ASTRA-sim: System Layer



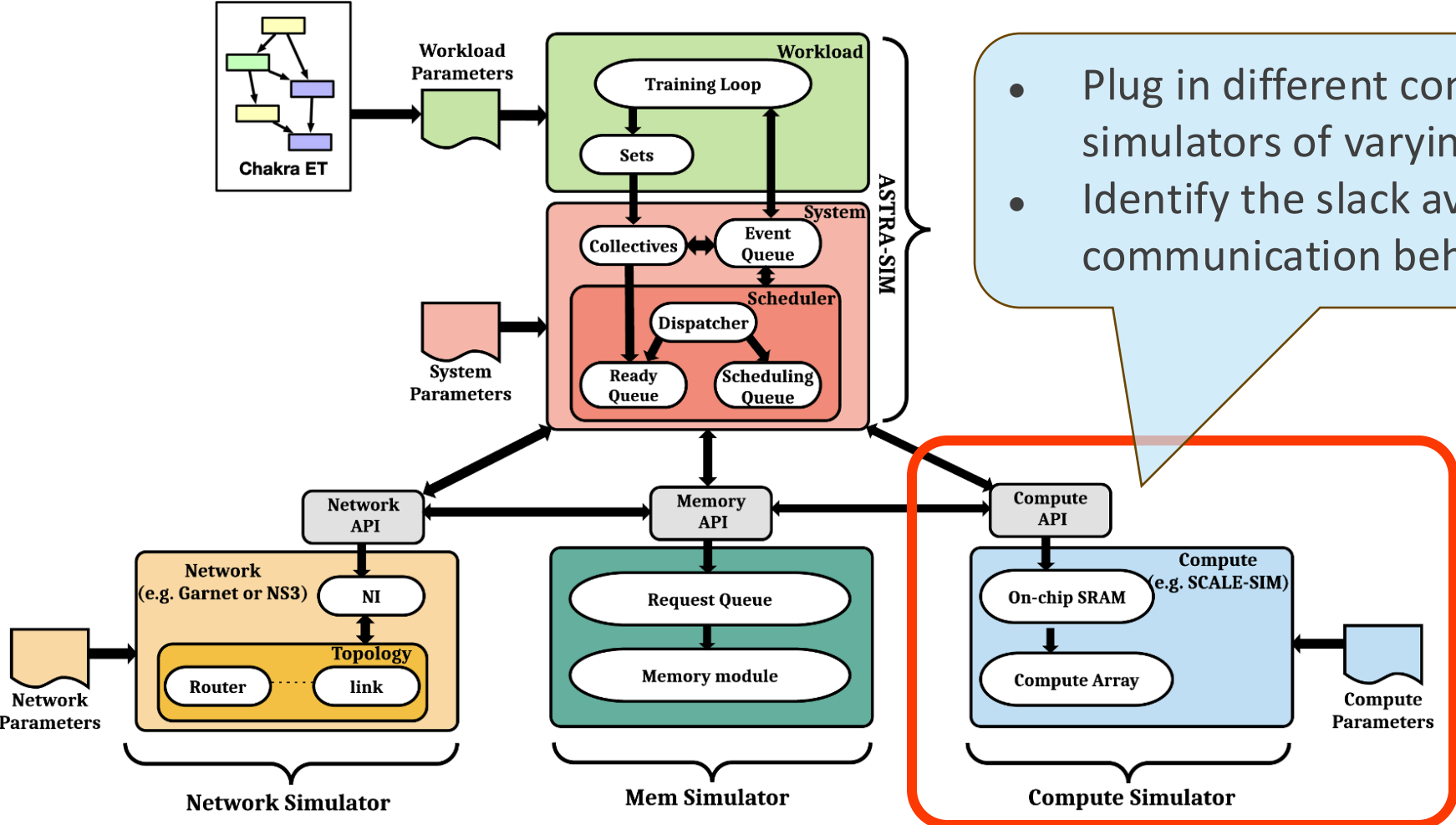
- Initiate compute, memory and collective operators
  - Collective algorithms are implemented using state machines.
- Pass events to compute, memory and network simulators

# ASTRA-sim: Network Layer



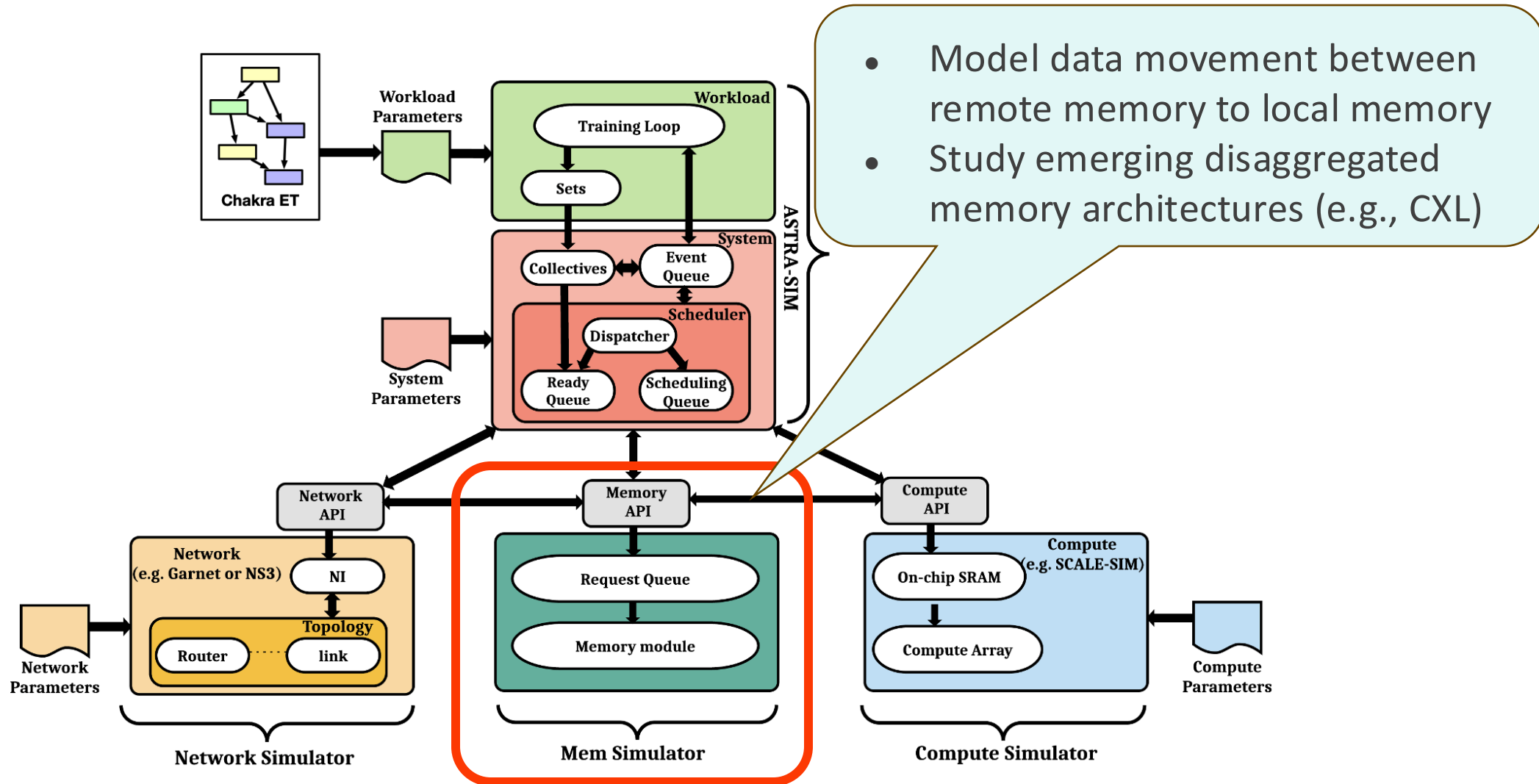


# ASTRA-sim: Compute Layer

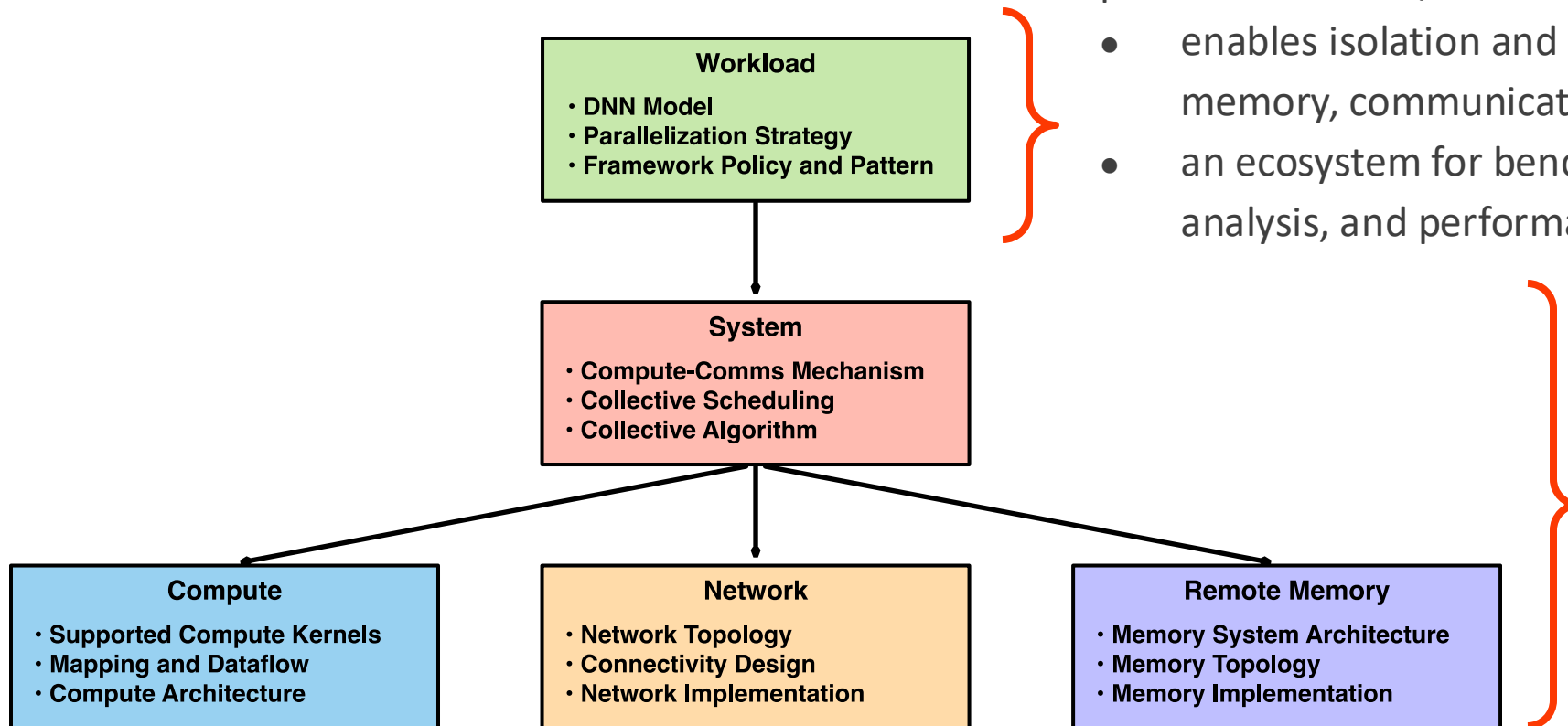


• Plug in different compute models / simulators of varying fidelity  
• Identify the slack available to hide communication behind compute

# ASTRA-sim: Memory Layer



# Conclusion



**Chakra Execution Trace:** an open graph-based representation of AI/ML workload execution

- enables isolation and optimization of compute, memory, communication behavior
- an ecosystem for benchmarking, performance analysis, and performance projection

**ASTRA-sim:** Distributed AI system simulator

- effectively models various aspects of distributed training
- allows mix-and-match of performance models for compute, memory and network (API-based)