Georgia Tech School of Electrical and Computer Engineering
College of Engineering

http://synergy.ece.gatech.edu

https://astra-sim.github.io

https://github.com/mlcommons/chakra

Enabling AI-driven Full-Stack Co-Optimization of Distributed AI Systems for the Architecture2.0 era

Tushar Krishna
Associate Professor, School of ECE
Georgia Institute of Technology
tushar@ece.gatech.edu

Semiconductor Research Corporation

CENTER FOR EVOLVABLE COMPUTING

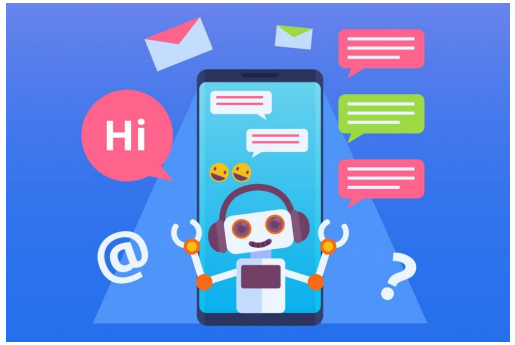
Semiconductor Research Corporation

CoCoSys
CENTER FOR THE CO-DESIGN OF COGNITIVE SYSTEMS

Architecture 2.0 Workshop @ ASPLOS 2026

AI is pervasive today!

Chatbots



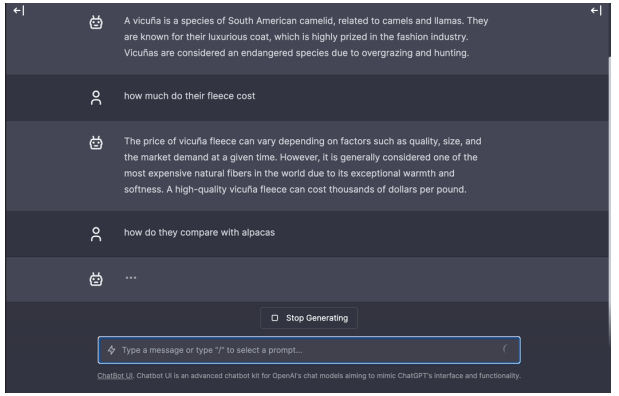
Code Generation

```ruby
def segmentation(items, separator)
  curr = []
  segments = []
  items.each do |item|
end

segmentation([1,2,4,0,2,5,0,3,0], 0).each do |segment|
  puts(segment.join(", "))
end
```

"25% of code at google in last quarter was AI generated "

Text Generation



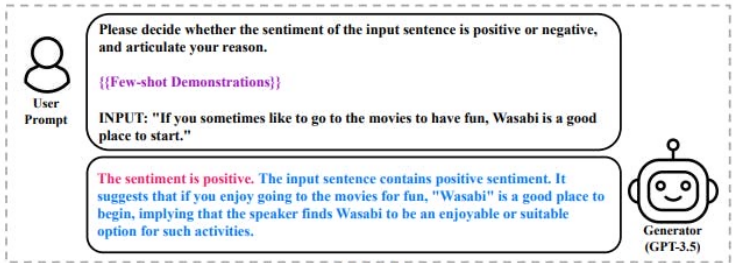
Language Translation

[*Instruction*]: Translate the following sentences from English to Chinese.
[*Input*]: Did you see it go?

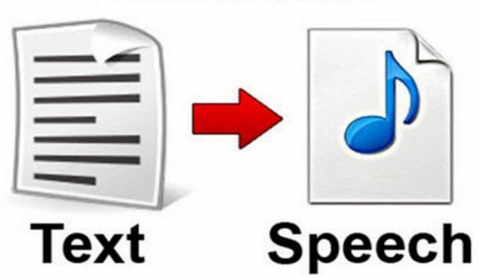
LLM

[*Output*]: 看清楚了吗？

Sentiment Analysis

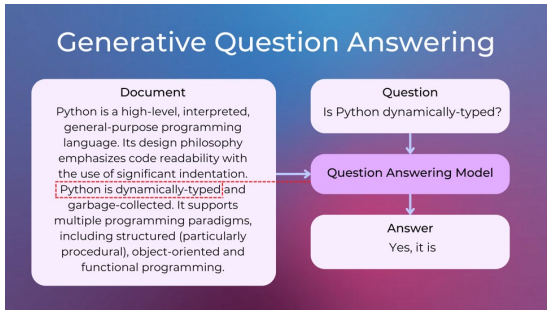


Text to Speech



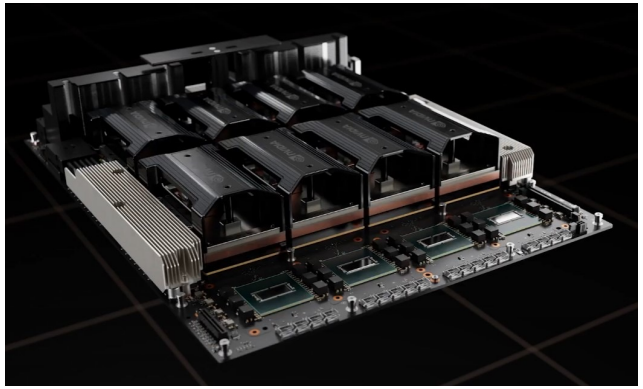
Recommendations

Question Answering



Algorithmic view of AI (Datasets and Models)

Computer Architect's view of AI



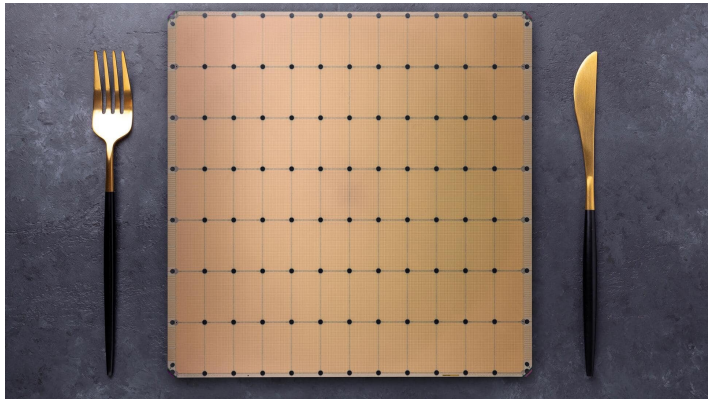
NVIDIA
HGX-H100 SuperPod

Google Cloud
TPUv4

SambaNova
SN40L

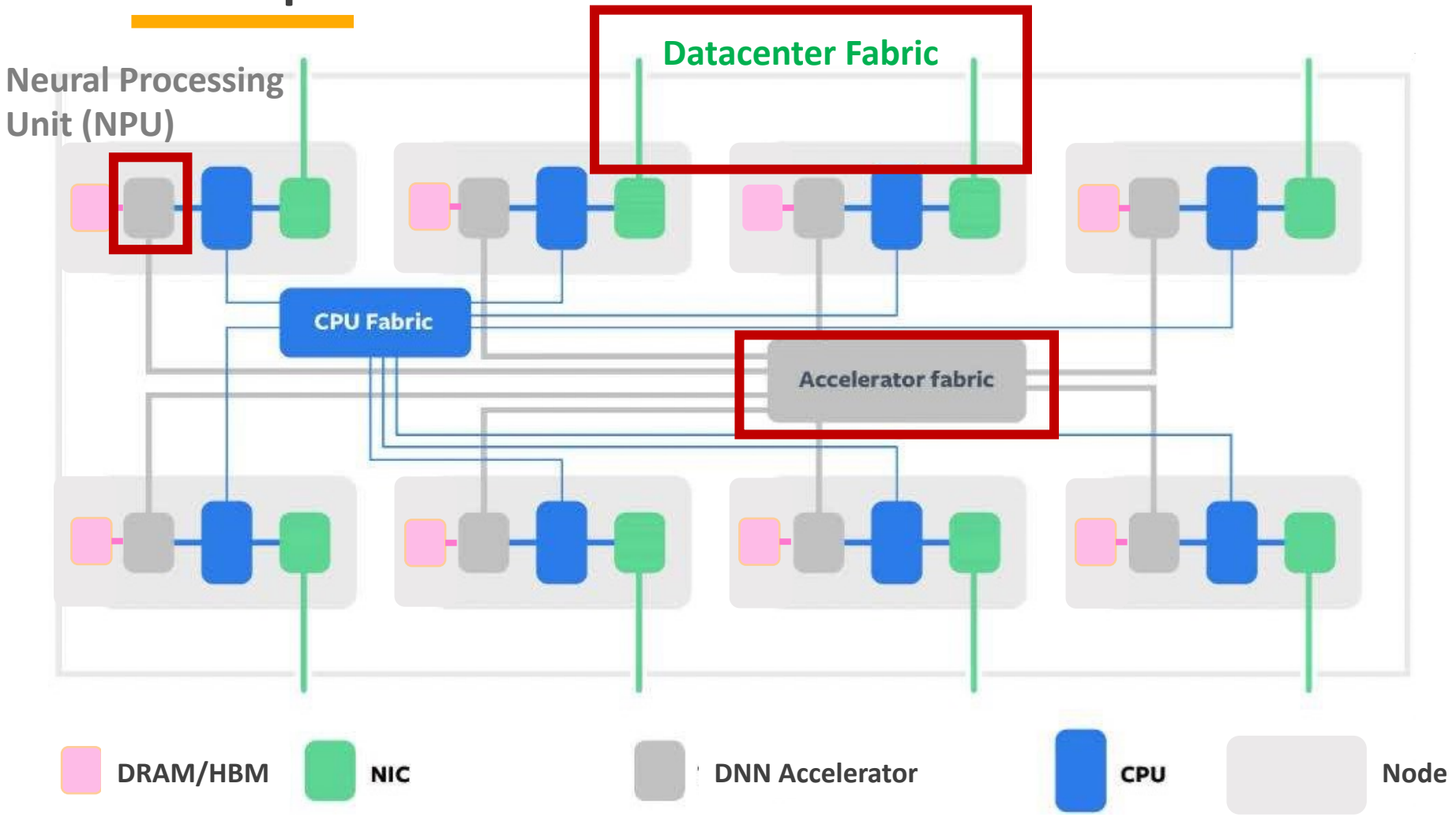
AMD
Instinct Platforms

Intel
Gaudi



Cerebras
Andromeda

Computer Architect's view of AI



- ✓ Customized accelerators for AI (aka NPUs)
- ✓ Customized network fabrics to scale the AI task across multiple accelerators

Why?

*Figure modified from "**Zion: Facebook Next- Generation Large Memory Training Platform", Misha Smelyanskiy, Hot Chips 31**"*

"Large" Language Models

Hundreds of ZettaFLOPs of compute

Model (Company)	Company	#Parameters [Billion]	Model Footprint (Assuming 2B/Param)	Training Footprint (Assuming 16B/Param*)
Clause 3 Opus	Anthropic	**2,000**	4.00 TB	32.00 TB
GPT-4	OpenAI	**1,760**	3.52 TB	28.16 TB
Gemini 1.5 Pro	Google	**1,500**	3.00 TB	24.00 TB
Samba-1	SambaNova	**1,400**	2.80 TB	22.40 TB
Cerebras-1T	Cerebras	**1,000**	2.00 TB	16.00 TB
Grok-3	xAI	**928**	1.86 TB	14.85 TB
DeepSeep-R1	DeepSeek-AI	**685**	1.37 TB	10.96 TB
PaLM	Google	**540**	1.08 TB	8.64 TB

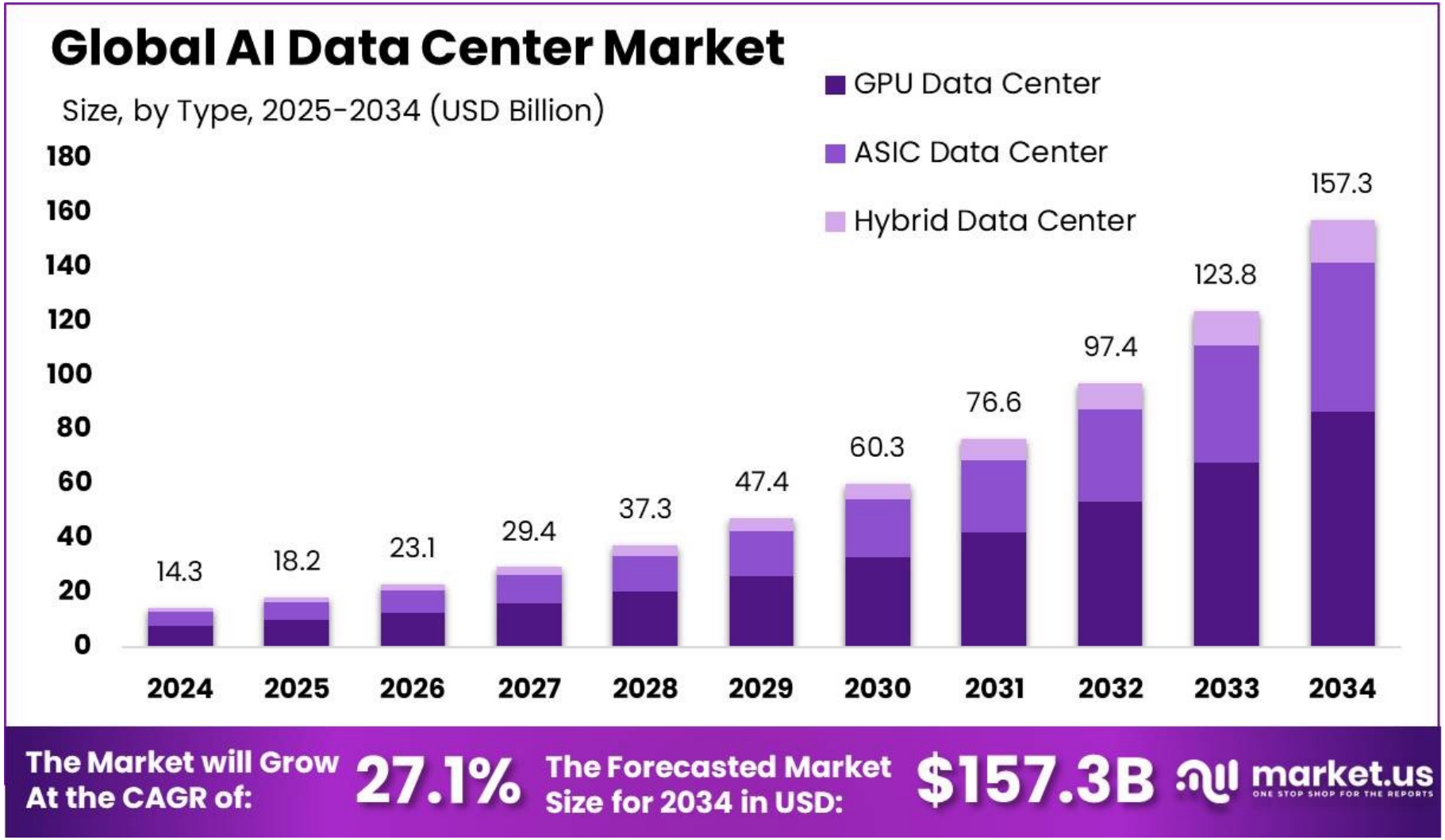
https://lifearchitect.ai/models/

**Assuming Mixed-precision Adam Optimizer (See Microsoft ZeRO)*

AI is a distributed systems problem!

	GPT-4 (2023)	GPT-5 (2025)
Total Parameters (Weights)	~1.8 Trillion (using MoE)	Unknown
Training Compute	~25000 NVIDIA A100 GPUs over 90-100 days	~170,000 H100/H200 GPUs over 2 years
Training Data	~13 Trillion Tokens	~70 Trillion Tokens
Inference Compute	128 NVIDIA A100 GPUs	Multiple GB200 (72 GPU) nodes
Context Length	32,000 Tokens	400,000 Tokens

The AI datacenter market continues to grow!



The AI datacenter “scale” continues to grow!

Google The Keyword Home Product news Company news Feed

TECHNOLOGY › RESEARCH

Nov 04, 2025

Meet Project Suncatcher, a research moonshot to scale machine learning compute in space.

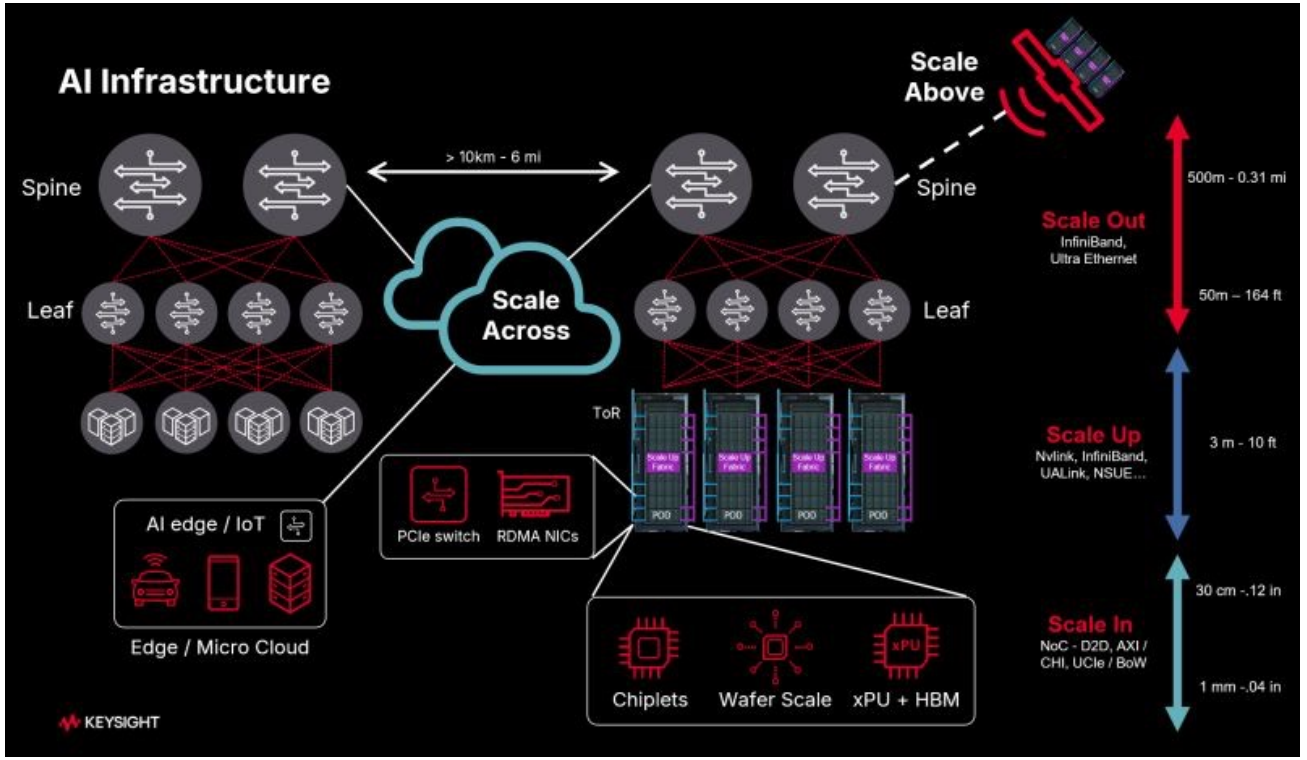
Artificial intelligence is a foundational technology that could help us tackle humanity's greatest challenges. Now, we're asking where we can go next to unlock its fullest potential. Today we're announcing Project Suncatcher, our new research moonshot to one day scale machine learning in space. Working backward from this potential future, we're exploring how an interconnected network of solar-powered satellites, equipped with our Tensor Processing Unit (TPU) AI chips, could harness the full power of the Sun.

Inspired by other Google moonshots like autonomous vehicles and quantum computing, we've begun work on the foundational work needed to one day make this future possible. We're excited that this is a growing area of exploration, and our initial research, shared today in a preprint paper, describes our approach to satellite constellation design, control, and communication, and also our initial learnings from radiation testing Google TPUs.

Our next step is a learning mission in partnership with Planet to launch two prototype satellites by early 2027 that will test our hardware in orbit, laying the groundwork for a future era of massively-scaled computation in space.

We are having to invent new terminology!

Scale-In → Scale-Up → Scale-Out → Scale-Across → Scale-Above



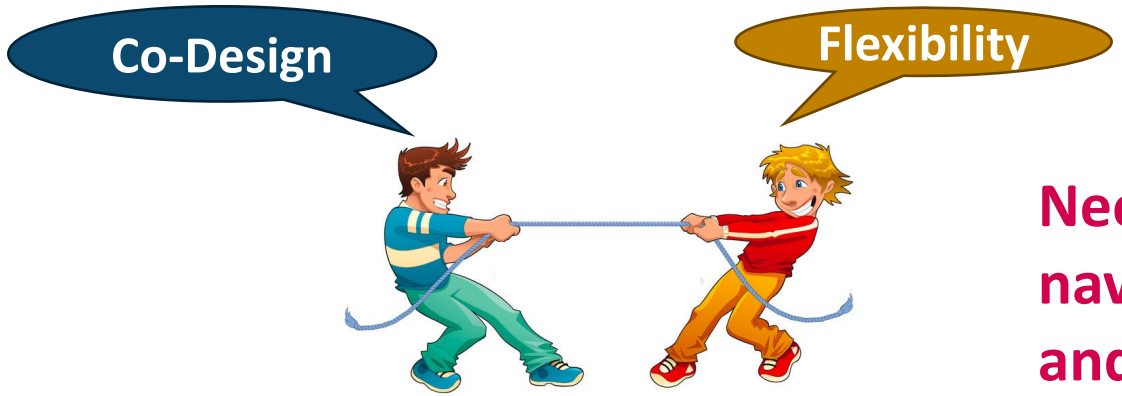
How to design and optimize AI Datacenters?

*"AI is not just about algorithms and software; it's about creating the right hardware infrastructure that can enable these algorithms to run efficiently. Hardware and software must be **co-designed** to fully unleash the potential of AI."*

— **Jensen Huang**, CEO of NVIDIA

*"The future of computing is going to be about **flexibility** — flexibility to create new architectures, flexibility to scale, flexibility to bring AI into every single thing that we do."*

— **Jensen Huang**, CEO of NVIDIA



Need an agile mechanism to navigate the HW-SW stack and make judicious bets

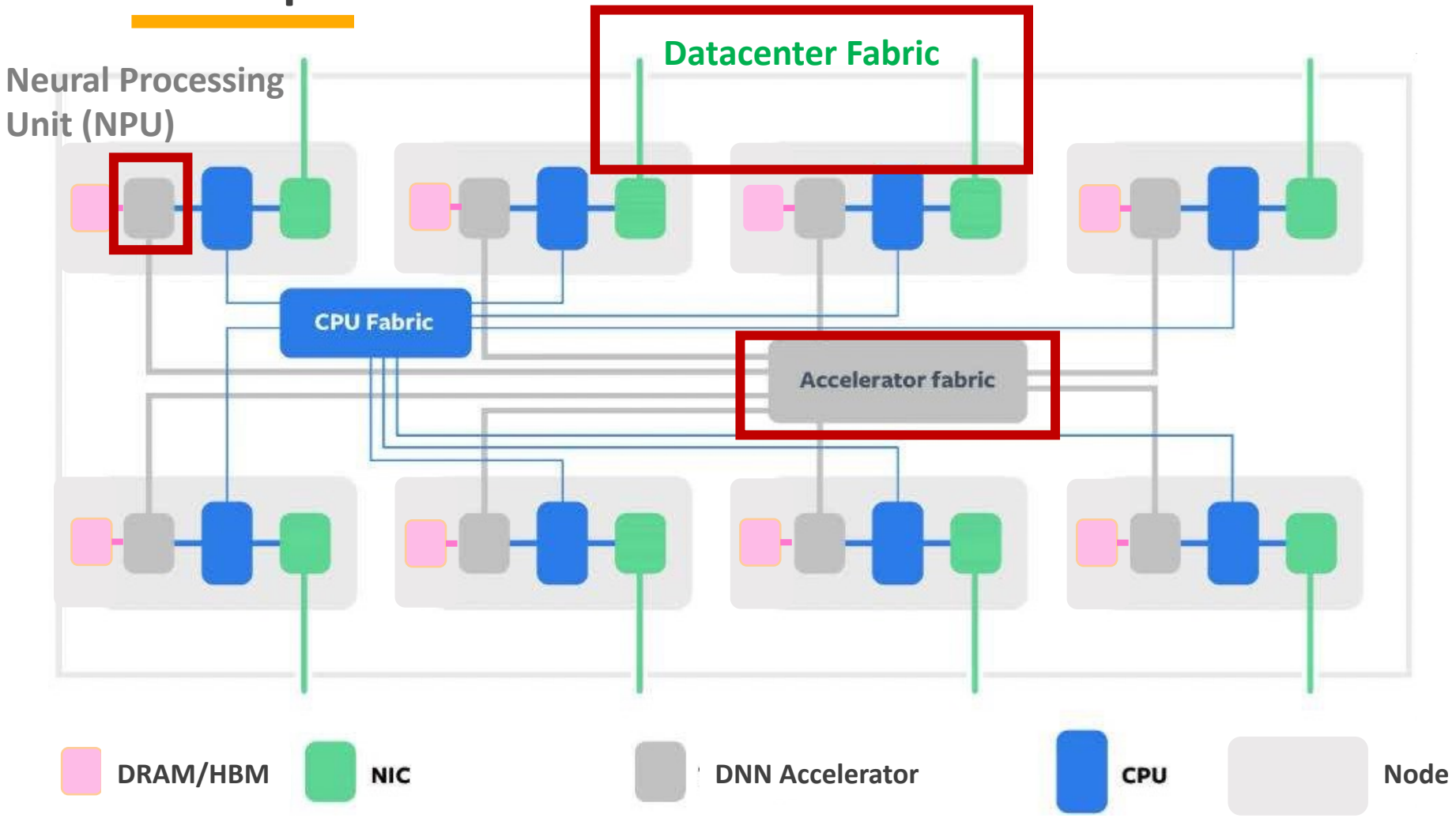
Outline

- Design Space of AI Platforms
- ASTRA-sim Ecosystem
- Case Study: Using AI to Navigate Search Space
- Conclusion

Outline

- **Design Space of AI Platforms**
- ASTRA-sim Ecosystem
- Case Study: Using AI to Navigate Search Space
- Conclusion

Computer Architect's view of AI

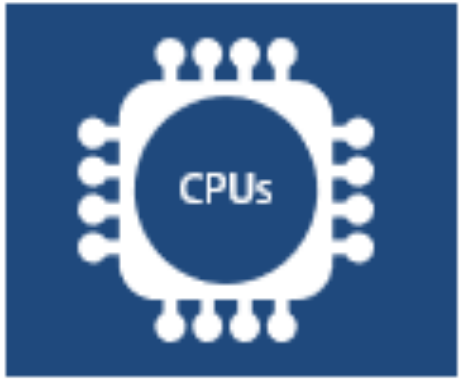


- ✓ Customized accelerators for AI (aka NPUs)
- ✓ Customized network fabrics to scale the AI task across multiple accelerators

*Figure modified from "**Zion: Facebook Next- Generation Large Memory Training Platform", Misha Smelyanskiy, Hot Chips 31**"*

Diverse Compute

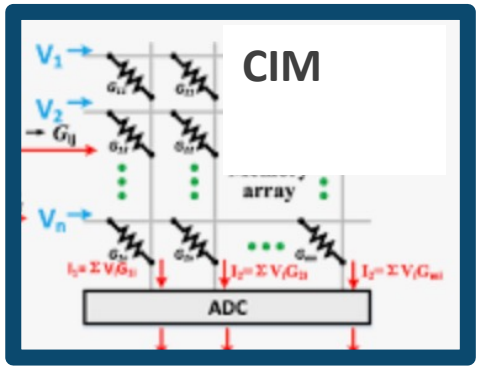
Intel
AMD



NVIDIA
AMD

Microsoft

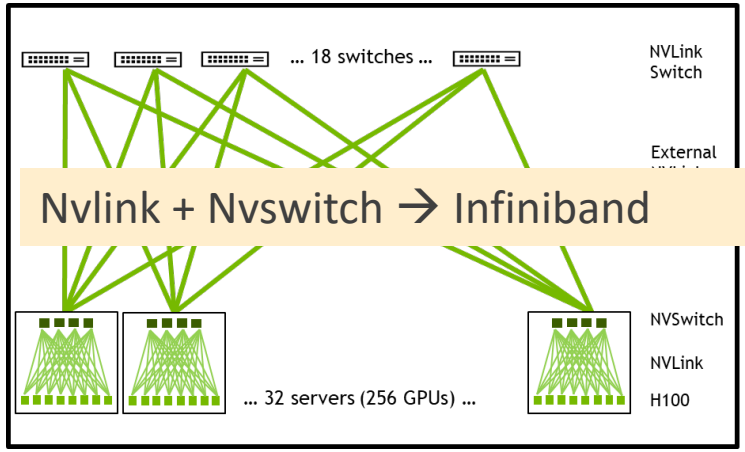
EnChargeAI



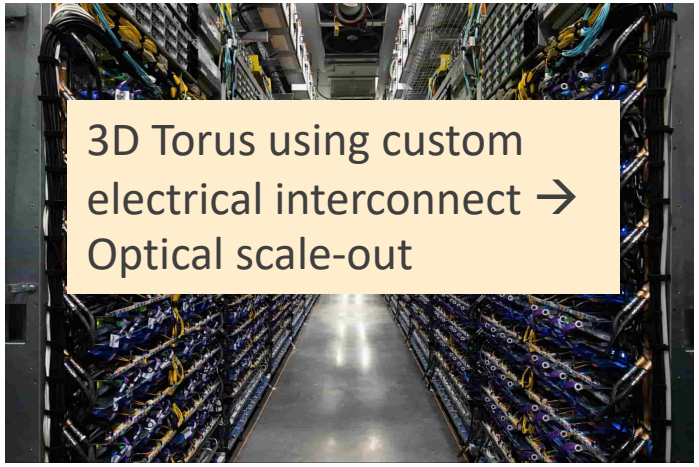
Google
Amazon
Meta
Microsoft
Groq
Cerebras
Rebellions

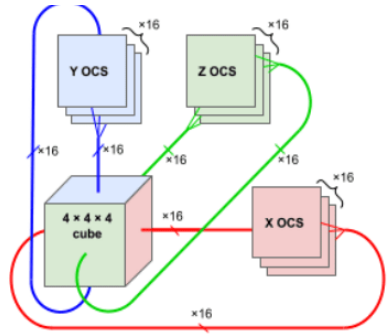
Diverse Networks

Scale-up → Scale-out

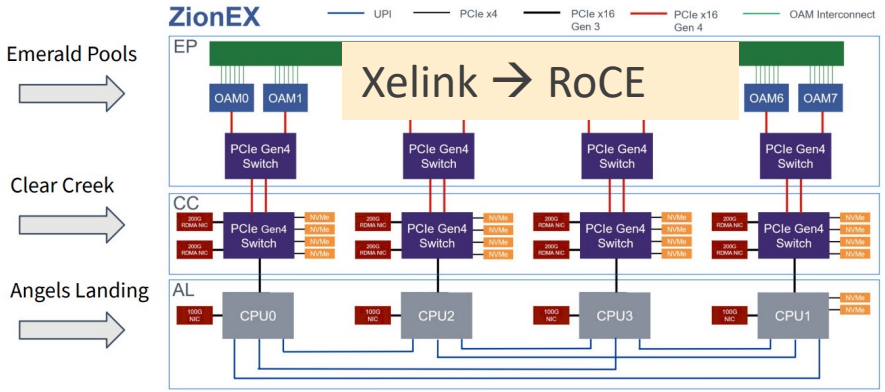


NVIDIA DGX

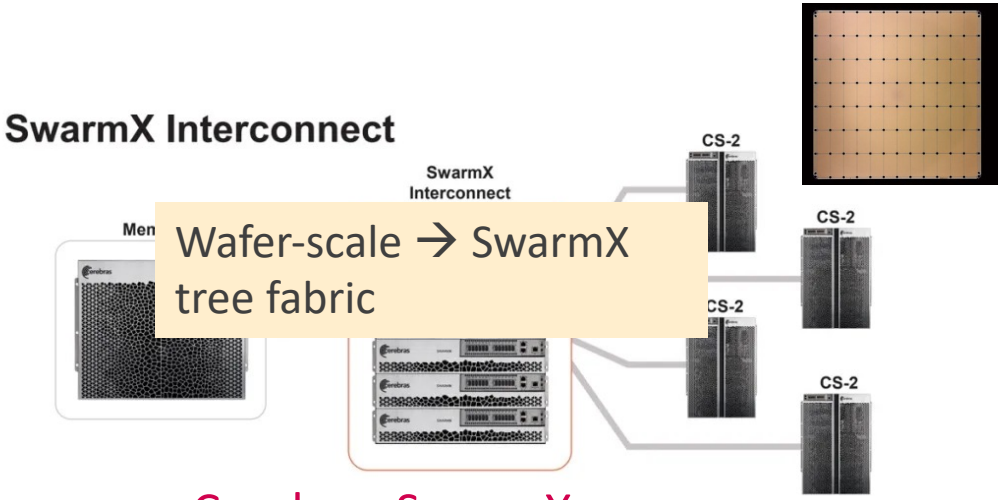




Google Cloud TPUv5

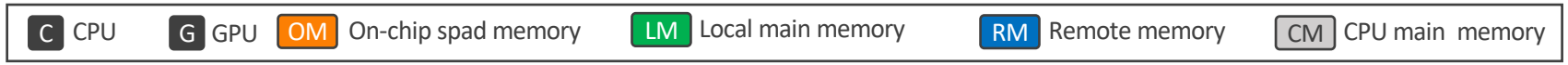


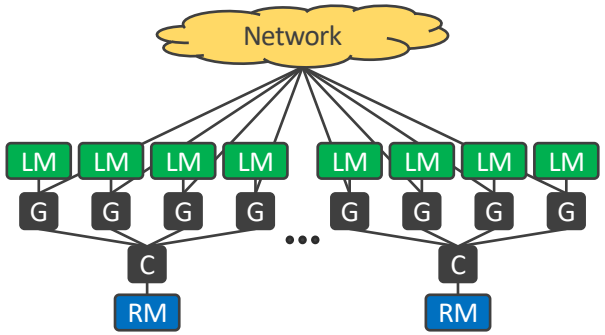
Intel Habana Gaudi



Cerebras SwarmX

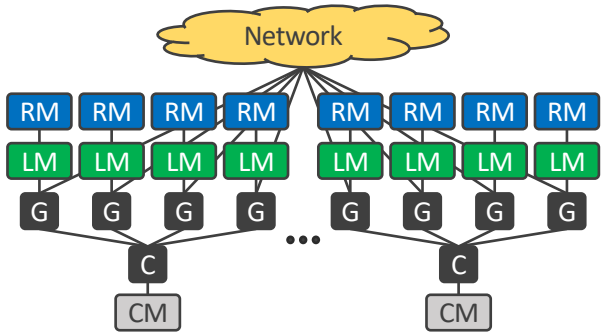
Diverse Memory Systems





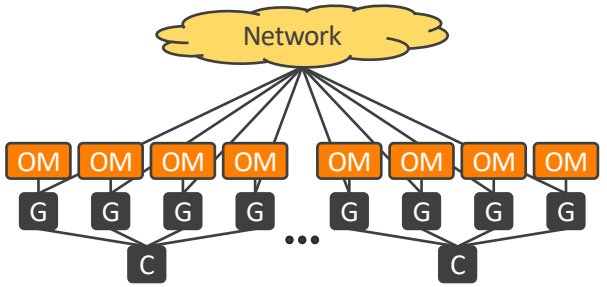
① Per-node memory expansion

Intel Habana, NVIDIA DGX2



② Per-GPU memory expansion

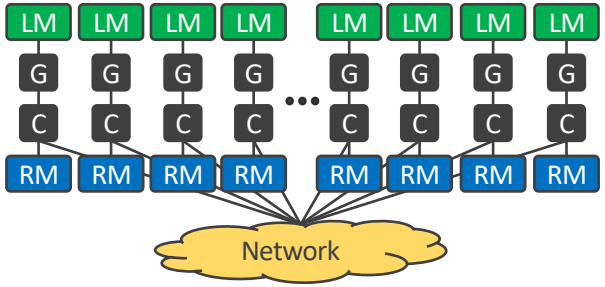
Astera Labs



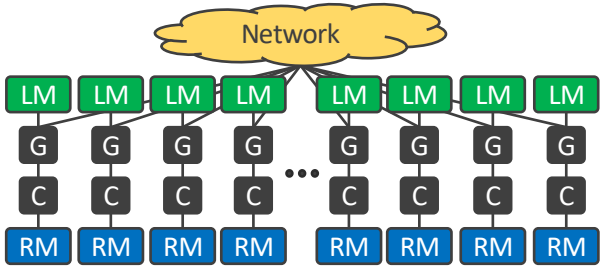
(5) On-chip (SRAM) memory

Groq

Cerebras CS-3



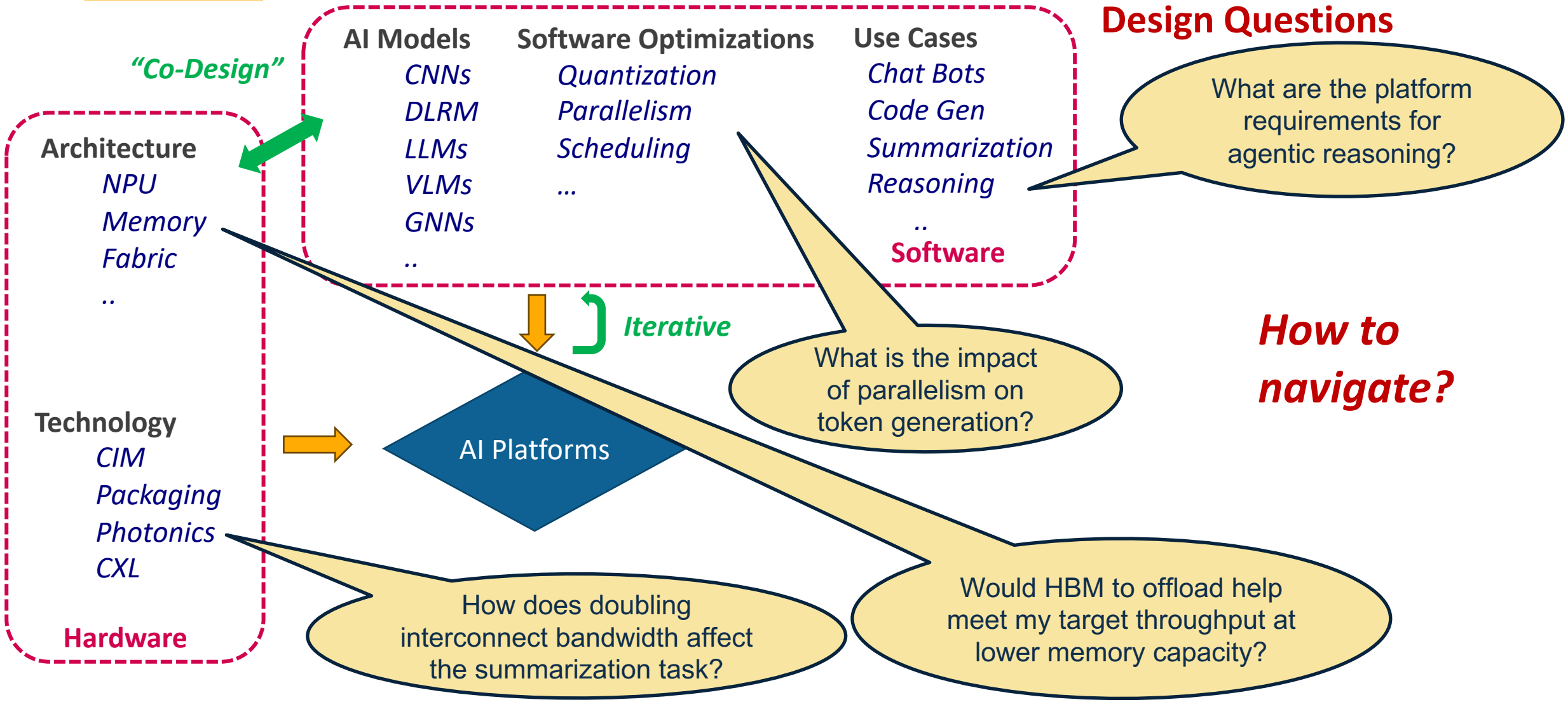
③ Integ. chip mem. exp with CPU network



④ Integ. chip mem. exp with GPU network

NVIDIA Grace Hopper

Cross-coupled Design-Space



Architecture2.0

Datasets

What datasets do we need? How should we collect these datasets for architecture research? What metadata should the datasets contain to enable broad usage? How do we create standard data formats from any ML algorithm?

Industry Relations

How do we share traces/infrastructure? What resources can industry offer? How can industry engage with academic efforts? How can academia transfer developed technologies to industry?

ML Algorithms

How can we learn and apply new ML algorithms to effectively design high-performance/efficient systems? How do we make our community more accessible to ML researchers? How do we embrace ML algorithm design as part of architecture research?

Workforce and Training

Can we create a systematic playbook for best known methods? How do we ensure strong baselines and reproducibility?

Best Practices

Can we create a systematic playbook for best known methods? How do we ensure strong baselines and reproducibility?

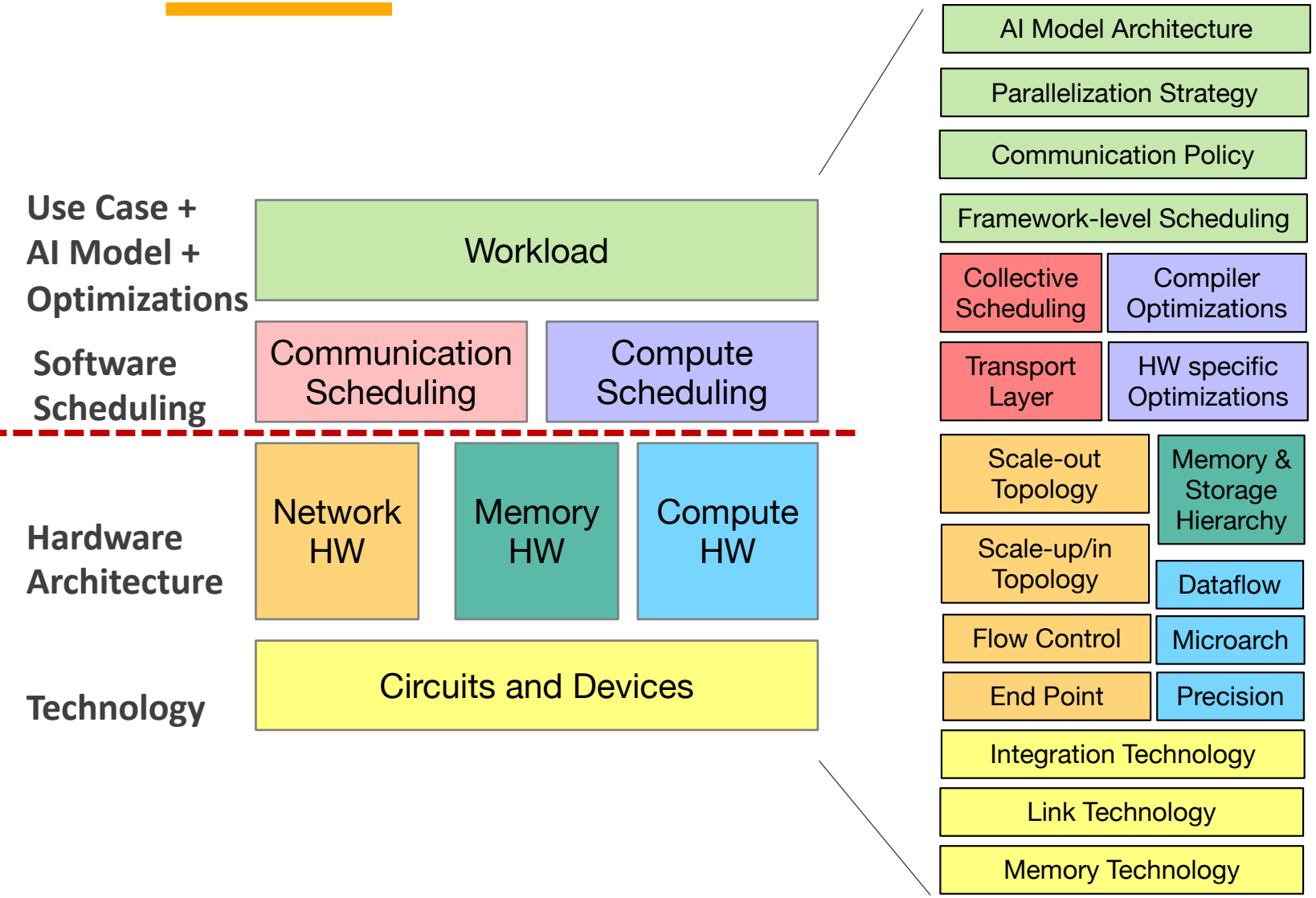
Tools and Infrastructure

What instrumentation mechanisms do we need for creating the datasets? What gym environments do we need to enable data-centric AI? How do we define standard data formats for interoperability?

Outline

- Design Space of AI Platforms
- **ASTRA-sim Ecosystem**
- Case Study: Using AI to Navigate Search Space
- Conclusion

Understanding the design-space



Co-Design Optimization Opportunities

LLM/ViT/Mamba/….

TP/PP/DP/FSDP/..

Collectives
Send-Rcv

Tiling
Layout
Mapping

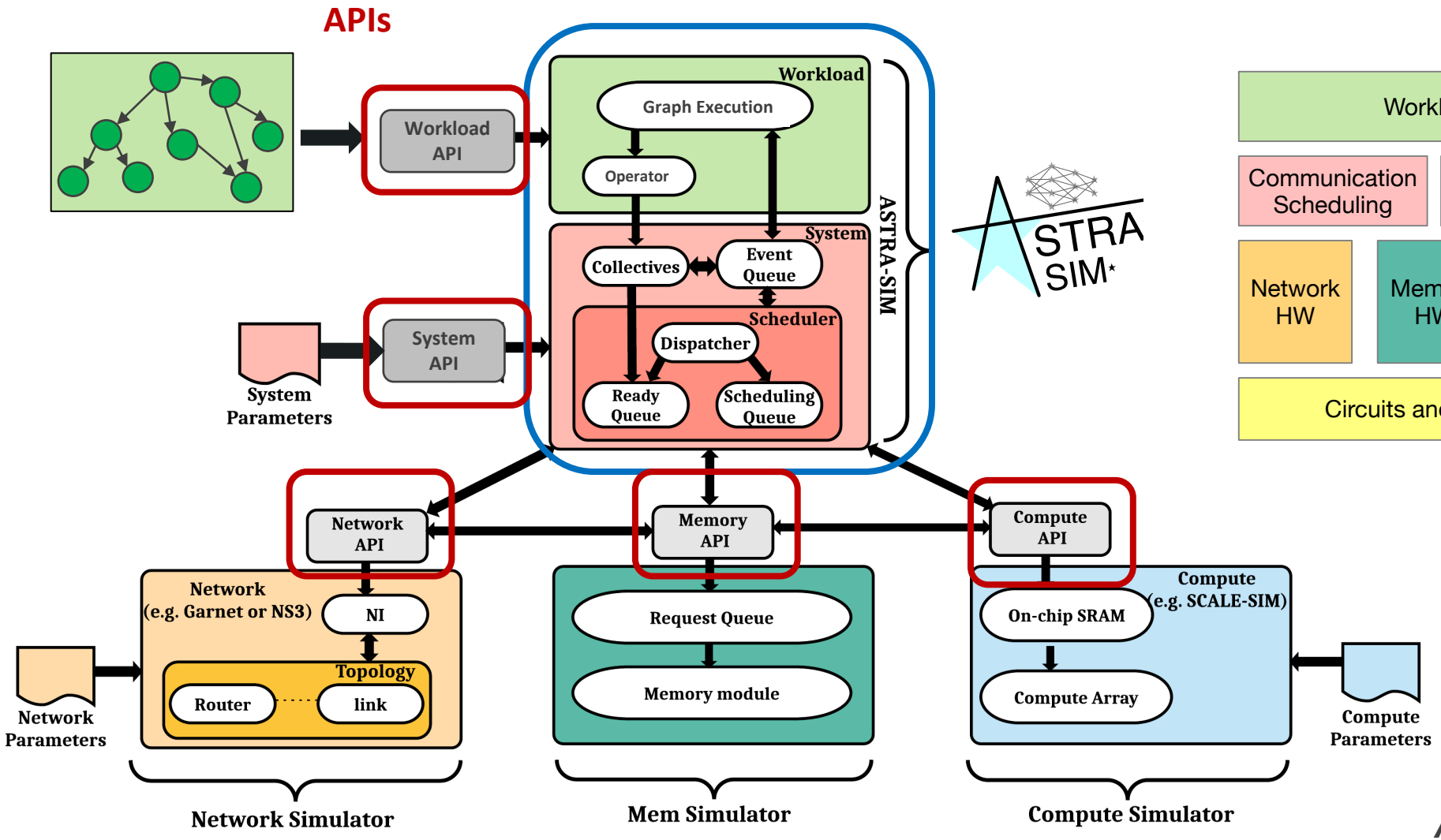
Topologies
Flow Control
Congestion Mgmt
Optimized Transport

Offload
CXL

Data Reuse
Sparsity

Frequency, Energy, Latency, Bandwidth

Introducing ASTRA-sim



ASTRA-sim: Design Principles

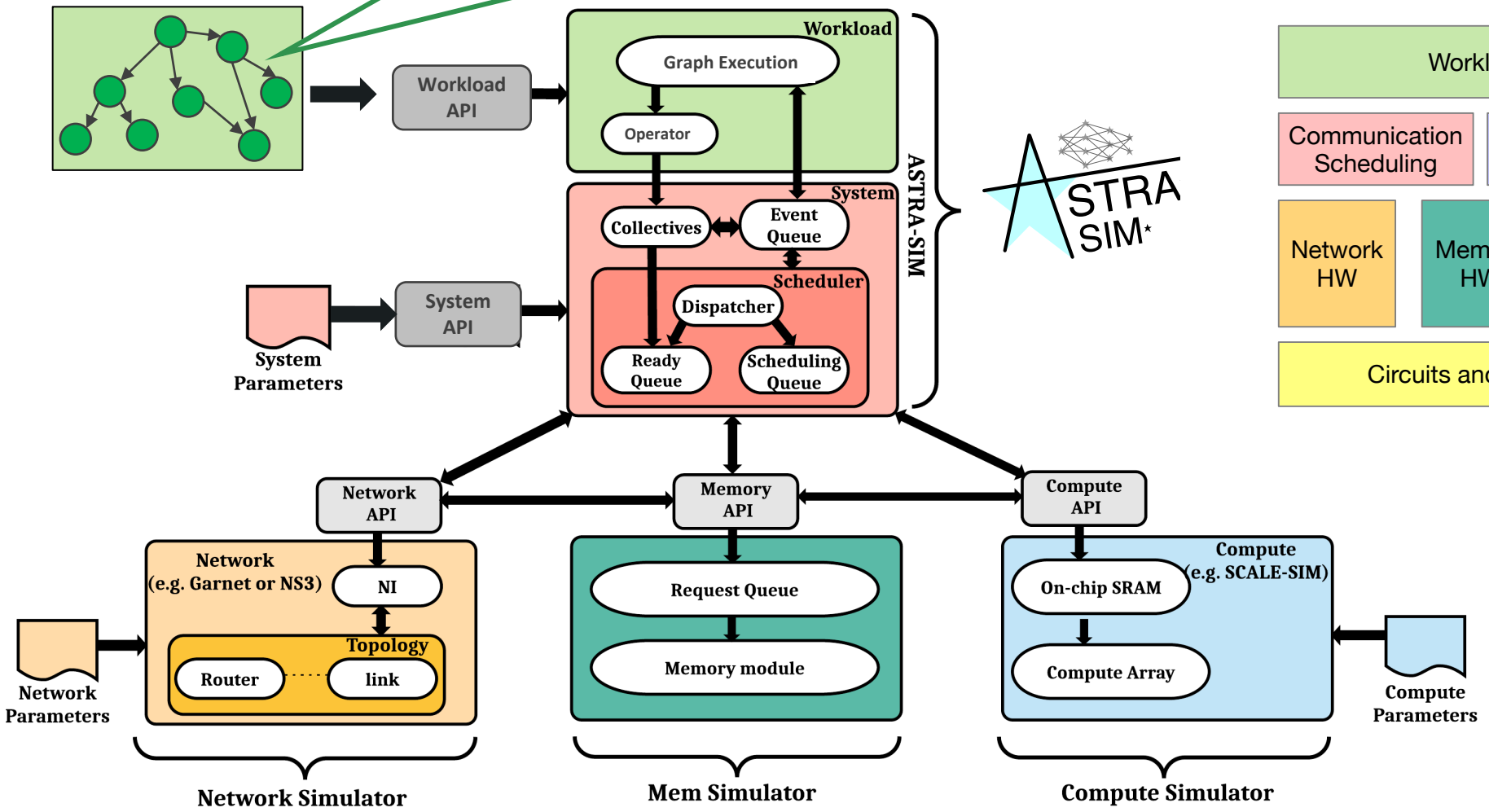
- User determines the model/simulator for compute/network/memory depending on the level of detail and simulation time they want

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

- **Reference Implementation:** http://github.com/astra-sim/astra-sim
- **Website:** https://astra-sim.github.io/
- **Tutorials:** https://astra-sim.github.io/tutorials

Workload

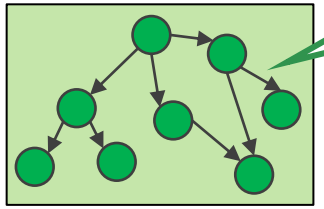
✓ Input + Model Architecture + Optimizations



ASTRA-sim Websit

Workload

✓ Input + Model Architecture + Optimizations



Training Workload = {Data Set, Model Architecture, System Optimizations}

LLM
MoE

Parallelism
Activation Recompute

Inference Workload = {Use case, Model Architecture, Model Optimizations, System Optimizations}

Chat Bot
Summarization
Q/A
Reasoning

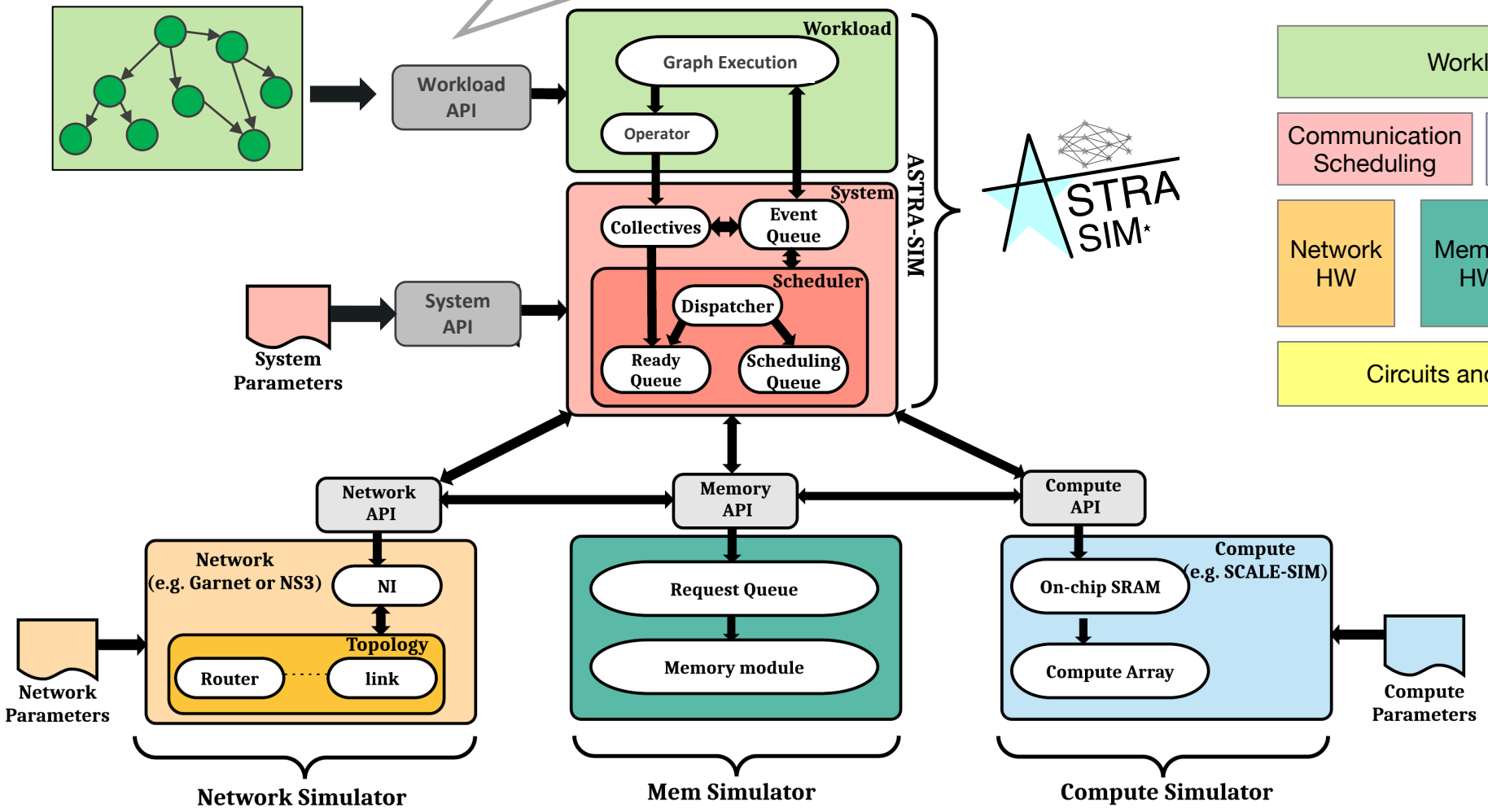
MQA/GQA
MoE
Sliding Window
Layer-wise KV sharing

Quantization
Weight Sparsity
KV pruning
Mixed precision

Flash Attention
Chunking prefill
Parallelism
Speculative Decoding

Workload API

Graphical representation of training/inference loop



ASTRA-sim Websit

Workload API

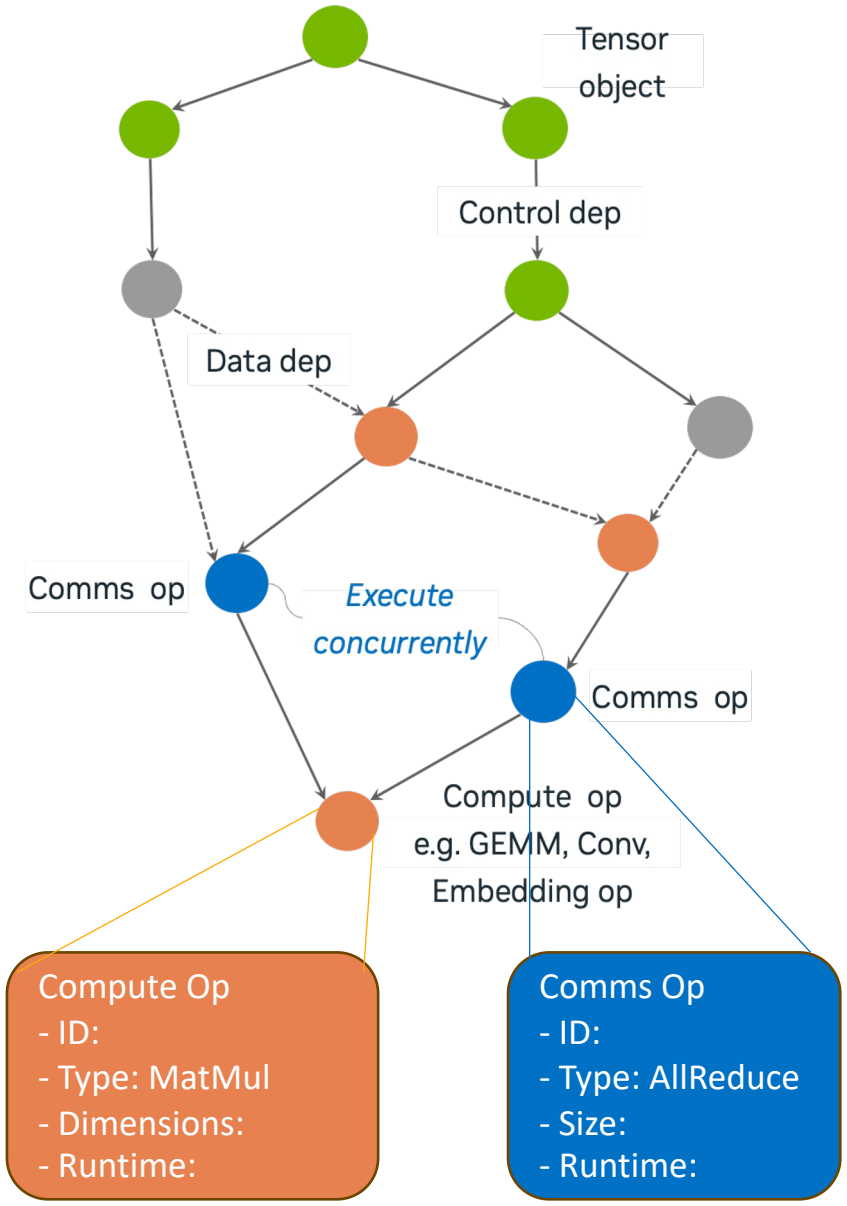
- Workload API captures **workload-specific characteristics**
  - DNN **Model**
  - **Parallelization** strategies
  - Control and Data **dependencies**
  - Compute and Communication **order**

- All workload characteristics are captured through MLCommons **Chakra Execution Trace** representation

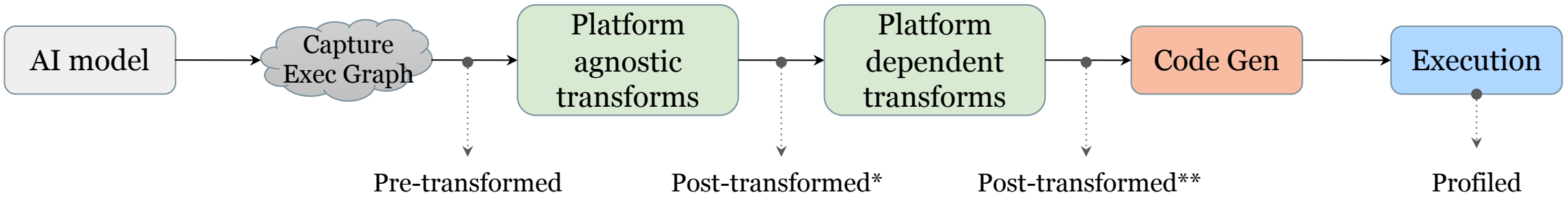
Chakra Execution Trace

- **Extensible and standardized graph format to represent AI workloads**
  - **Nodes:** primitive operators and tensor objects with attributes and timing
  - **Edges:** data and control dependency

- **Benefits**
  - Isolate comms and compute operators
  - Operator, dependencies, and timing for replay, simulation, and analysis
  - Flexible to represent both workloads and collective implementations
  - Graph transformations to obscure sensitive IP

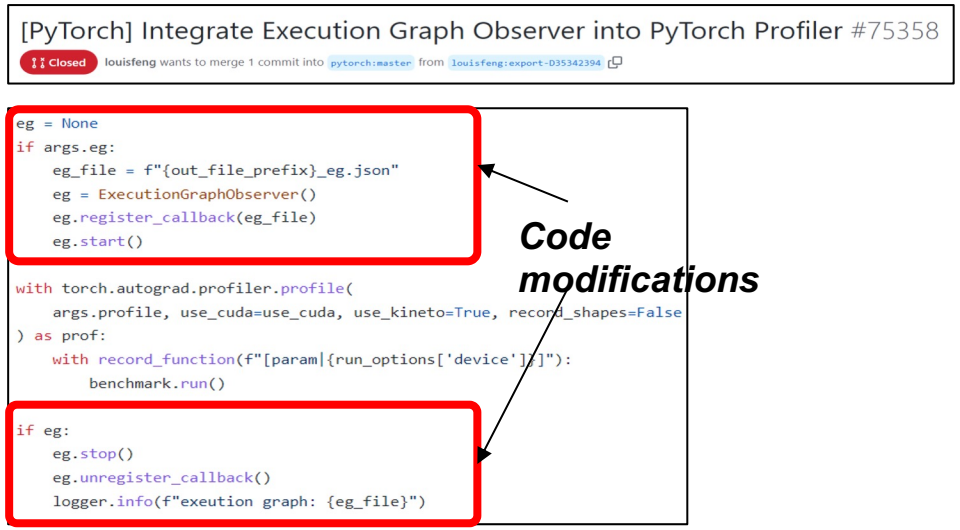


Chakra Traces: Collection and Synthesis



Type of Execution Traces

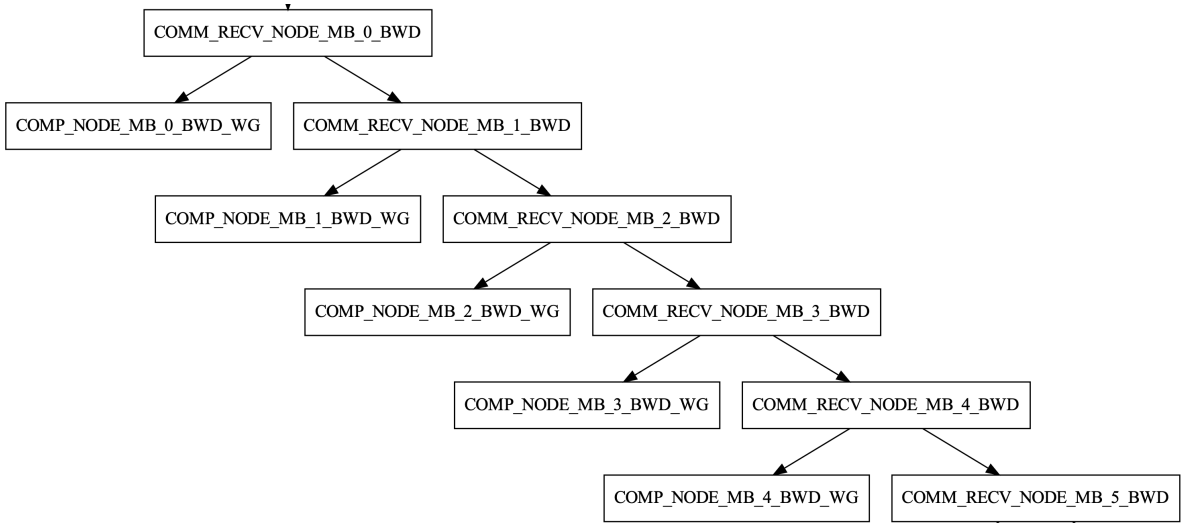
1. **Pre-transformed:** original model
2. **Post-transformed#:** optimized graph (may or may not be platform dependent)
   i.e., PyTorch2.0 FXgraphs
3. **Profiled:** graph executed on a specific platform
4. **Synthesized:** via analytical or statistical models

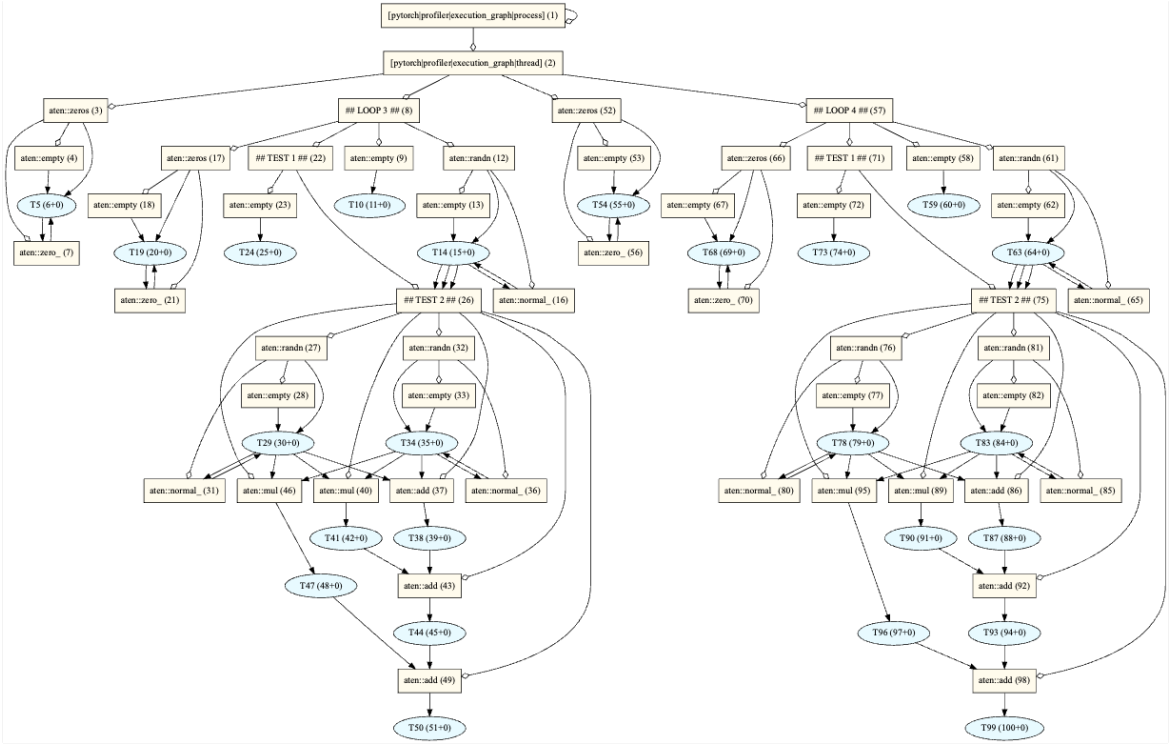


https://github.com/astra-sim/symbolic_tensor_graph

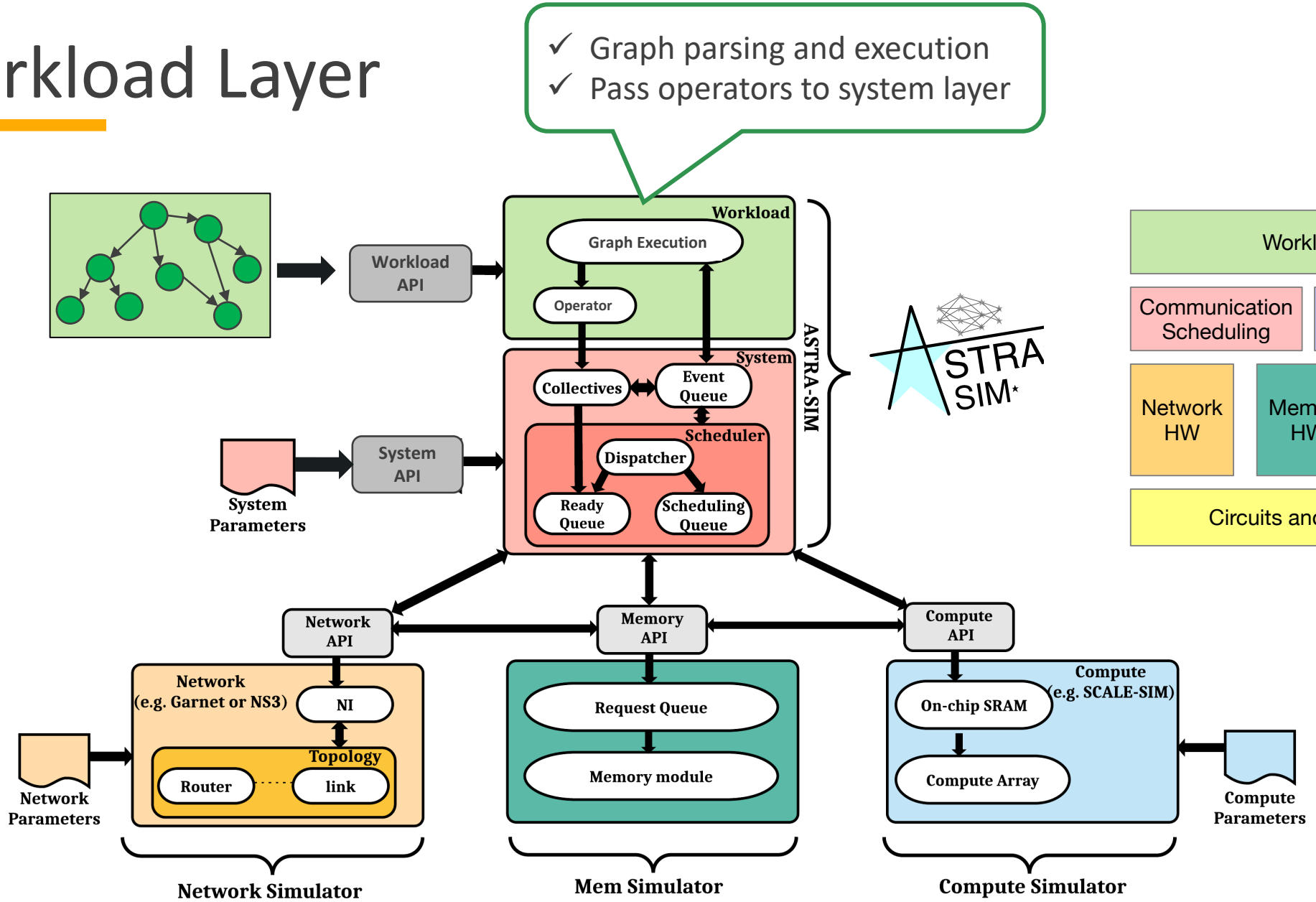
ICLR 2025 paper: https://arxiv.org/abs/2411.02322

Sample Chakra Traces

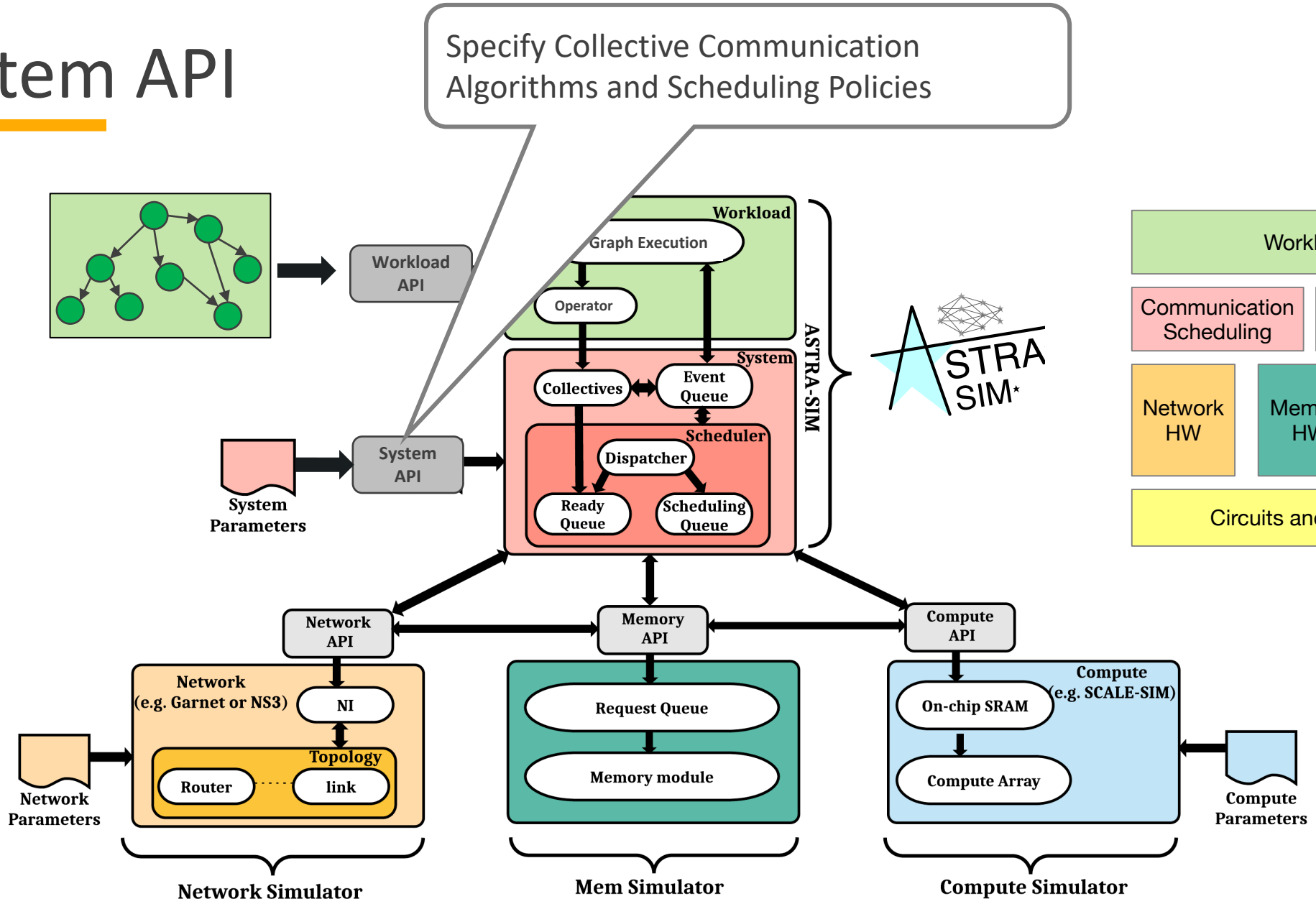




Workload Layer

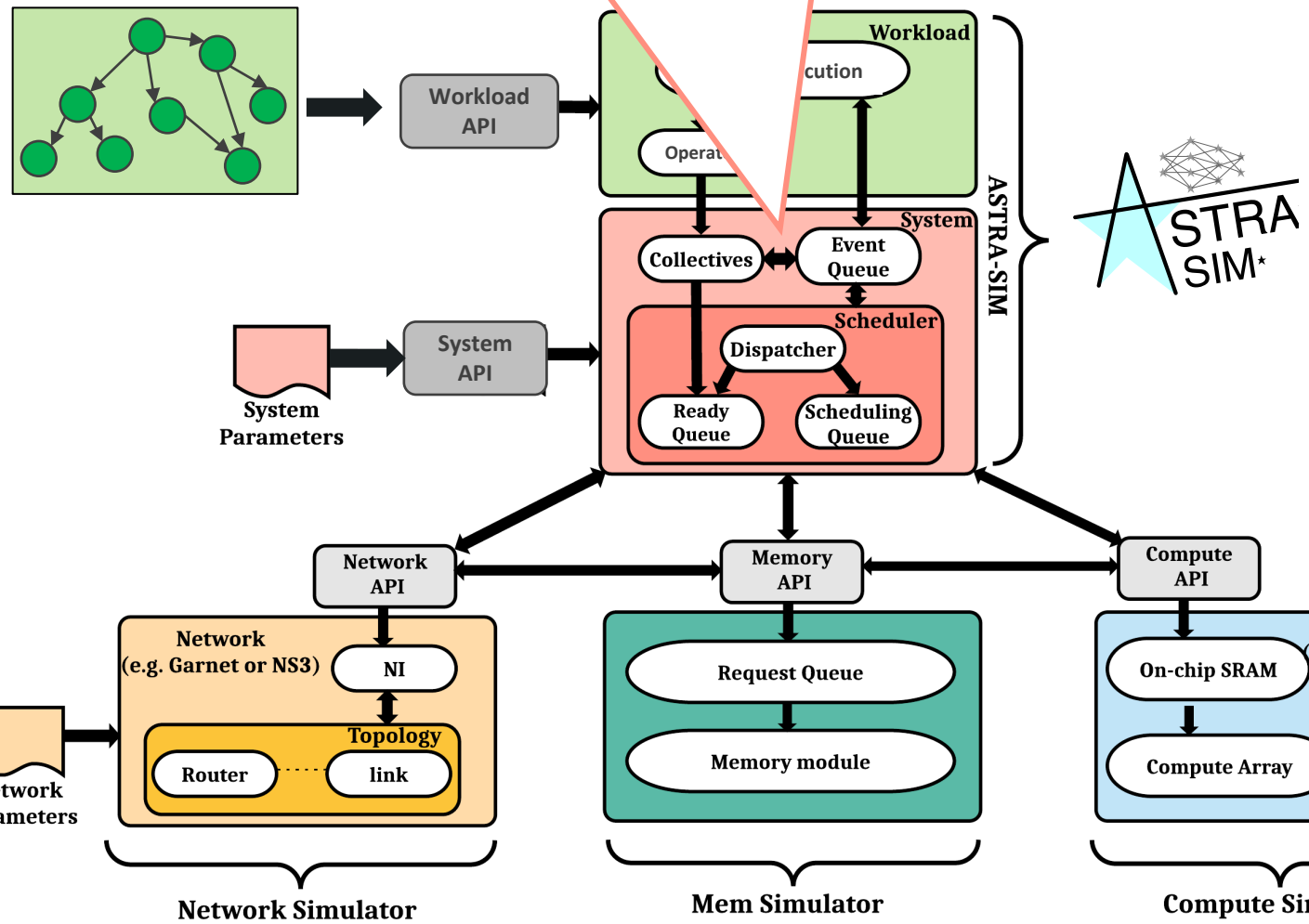


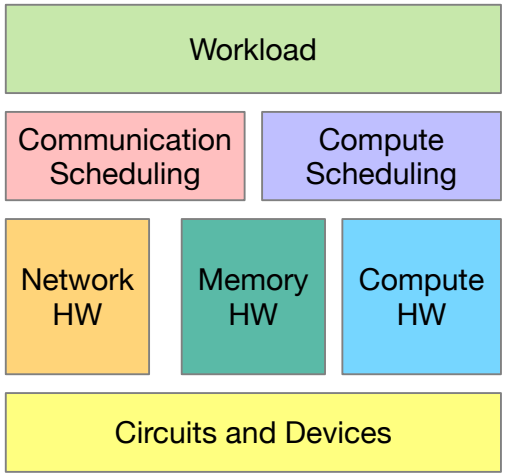
System API


Specify Collective Communication Algorithms and Scheduling Policies
Workload API
Workload
Graph Execution
Operator
System
Collectives
Event Queue
Scheduler
Dispatcher
Ready Queue
Scheduling Queue
ASTRA-SIM
System API
System Parameters
Network API
Memory API
Compute API
Network (e.g. Garnet or NS3)
NI
Topology
Router
link
Request Queue
Memory module
Compute (e.g. SCALE-SIM)
On-chip SRAM
Compute Array
Network Parameters
Compute Parameters
Network Simulator
Mem Simulator
Compute Simulator
Workload
Communication Scheduling
Compute Scheduling
Network HW
Memory HW
Compute HW
Circuits and Devices

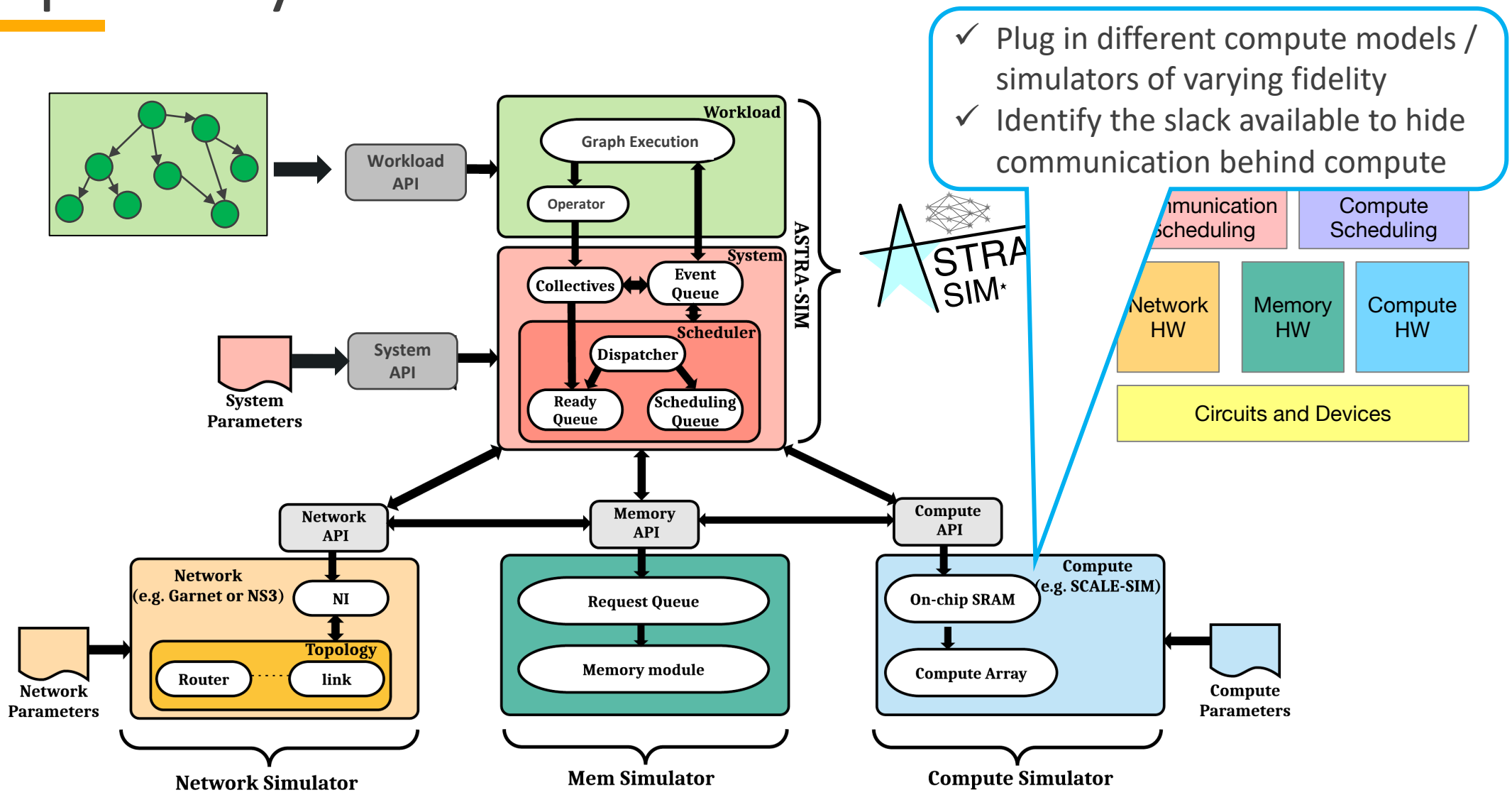
System Layer

- ✓ Identify compute, memory and collective operators
- ✓ Pass operators to compute, memory and network simulators



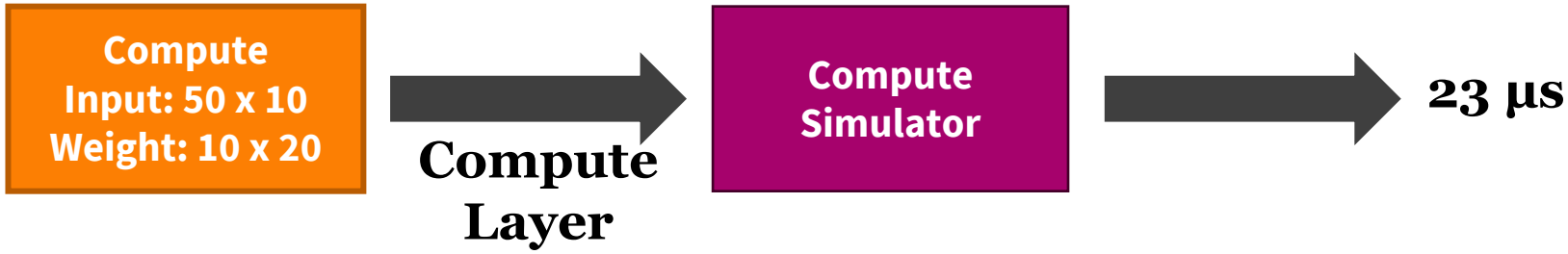


Compute Layer

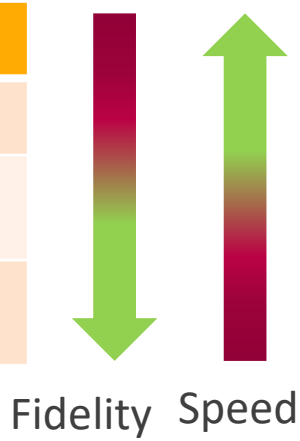


Compute Layer

- **Simulates compute times required** for Chakra ET's compute node



Model	Purpose	Notable Feature
Roofline	Analytical: First-order roofline	Fast analysis for compute vs memory boundness
SCALE-sim	Cycle-accurate: systolic array and memory	Models Google-TPUv5-like SoC
Real GPU**	Run compute operator on real GPU	Measured runtime

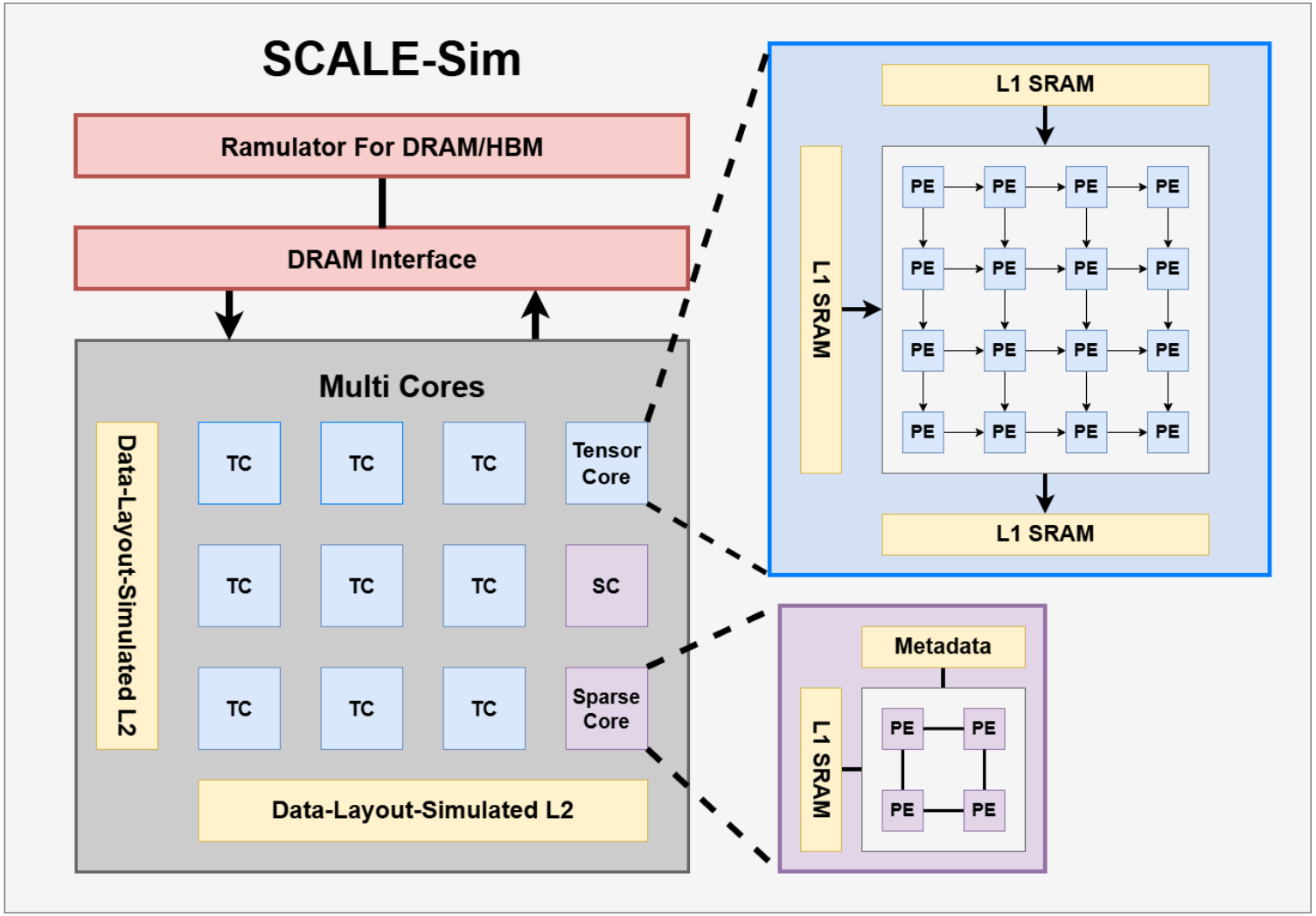


***In progress*

Example: SCALE-Sim NPU Simulator

Hardware Spec Configuration File

Wrokload Topology File

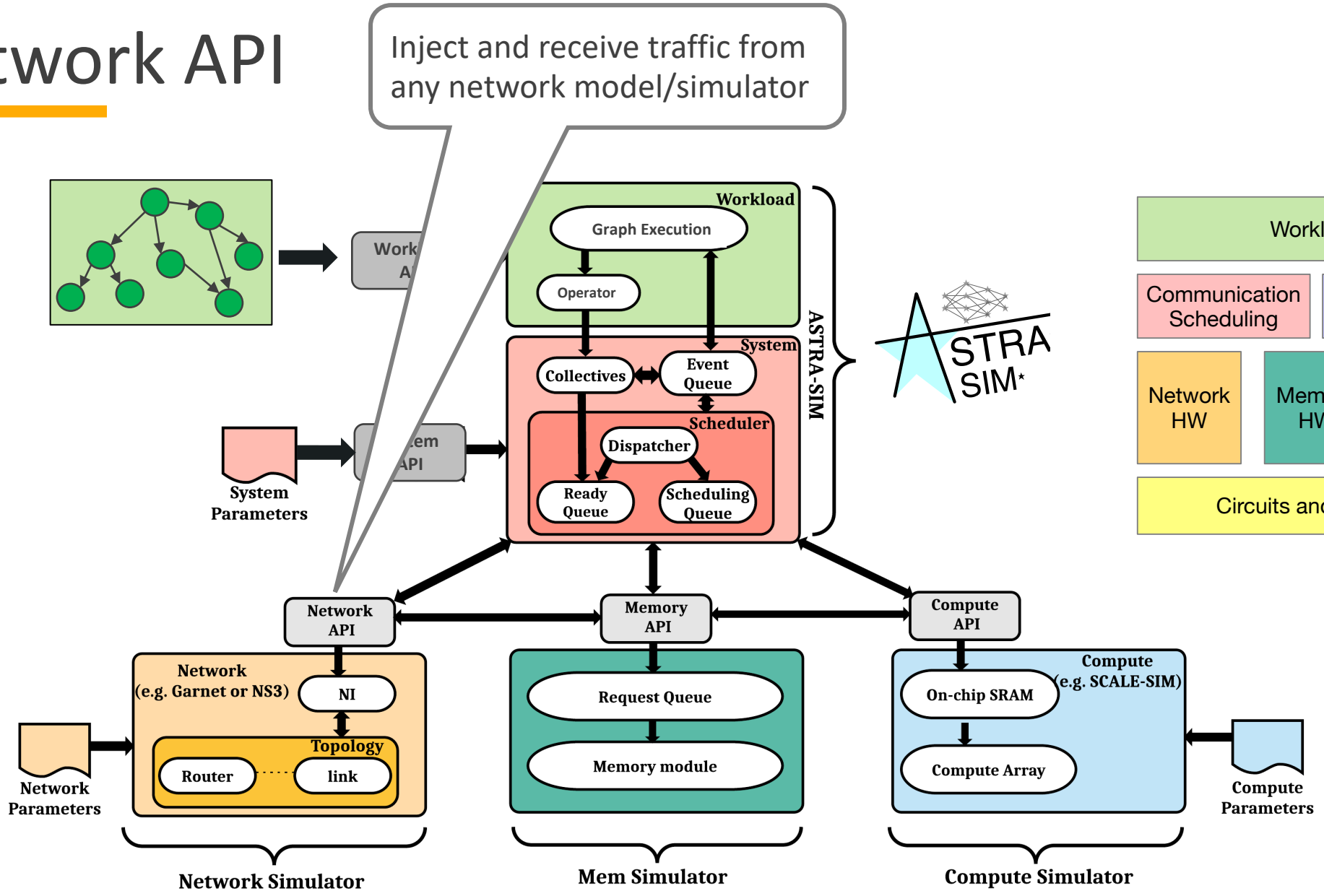


Repot: Cycles, Bandwise etc.

Cycle Accurate Traces

https://scalesim-project.github.io/

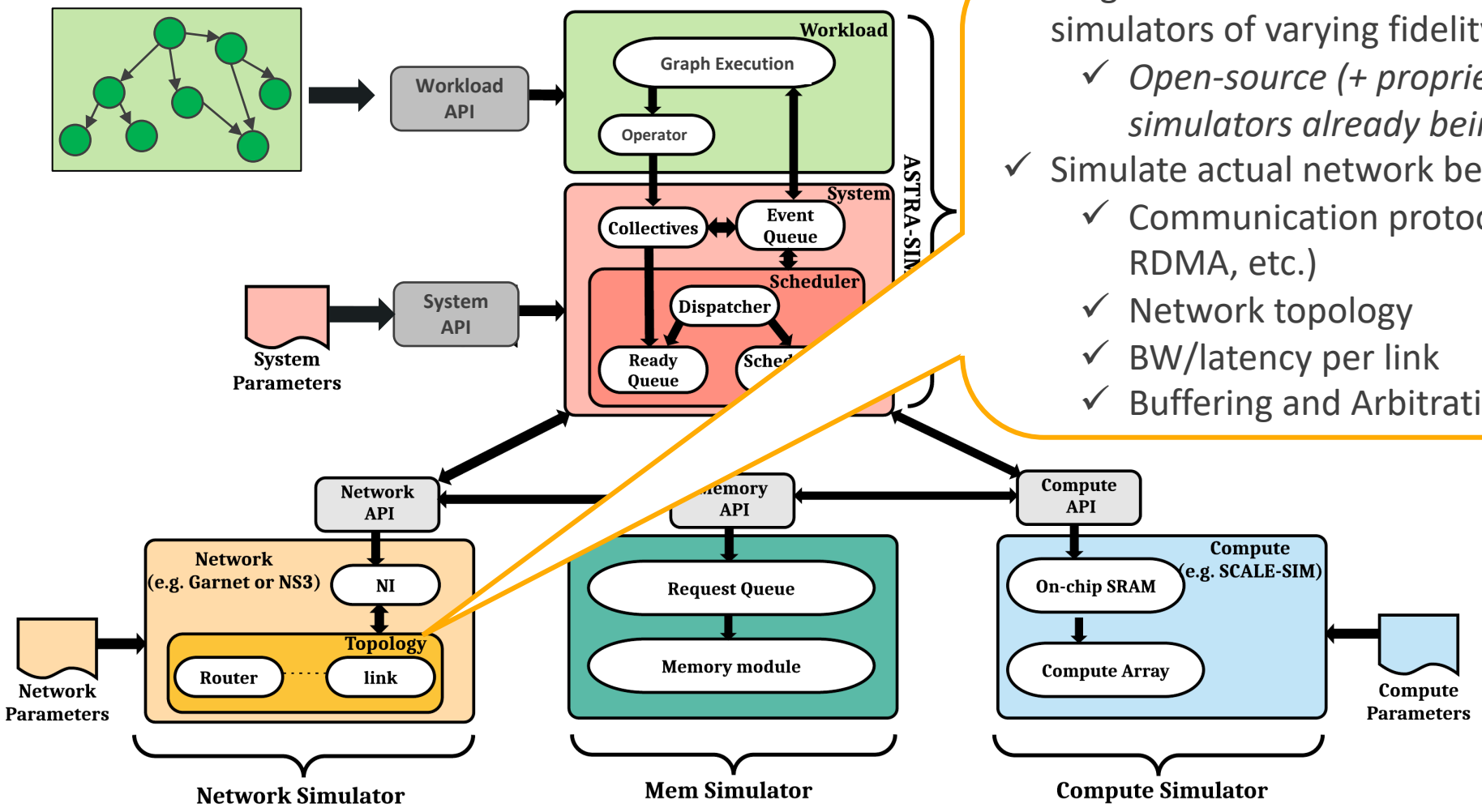
Network API


Inject and receive traffic from any network model/simulator
Workload
Graph Execution
Operator
System
Collectives
Event Queue
Scheduler
Dispatcher
Ready Queue
Scheduling Queue
ASTRA-SIM
System Parameters
Network API
Memory API
Compute API
Network (e.g. Garnet or NS3)
NI
Topology
Router
link
Request Queue
Memory module
Compute (e.g. SCALE-SIM)
On-chip SRAM
Compute Array
Network Parameters
Compute Parameters
Network Simulator
Mem Simulator
Compute Simulator
Workload
Communication Scheduling
Compute Scheduling
Network HW
Memory HW
Compute HW
Circuits and Devices
ASTRA-sim Websit

NetworkAPI

- **`sim_send(msg_size, src, dest, callback)`**
  - **Simulate sending a message** of size msg_size from src through dest and **invoke callback function** once transmission has finished
- **`sim_recv(msg_size, src, dest, callback)`**
  - **Simulate receiving a message** of size msg_size from src through dest and **invoke callback function** once transmission has finished
- **`sim_schedule(delta, callback)`**
  - Invoke callback function after delta time
- **`sim_get_time()`**
  - Return current time of simulation to the frontend

Network Layer



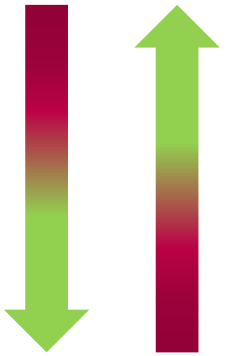
- ✓ Plug in different network models / simulators of varying fidelity
  - ✓ *Open-source (+ proprietary) simulators already being used*
- ✓ Simulate actual network behavior
  - ✓ Communication protocols (TCP, RDMA, etc.)
  - ✓ Network topology
  - ✓ BW/latency per link
  - ✓ Buffering and Arbitration

ASTRA-sim Websit

Network Layer

- Simulates **actual network behavior** (send/recv)
- Supports **multiple** network models/simulators through NetworkAPI
  - Enabling the simulation of various scales/fidelity
- We have released **4 network simulators** implementing NetworkAPI

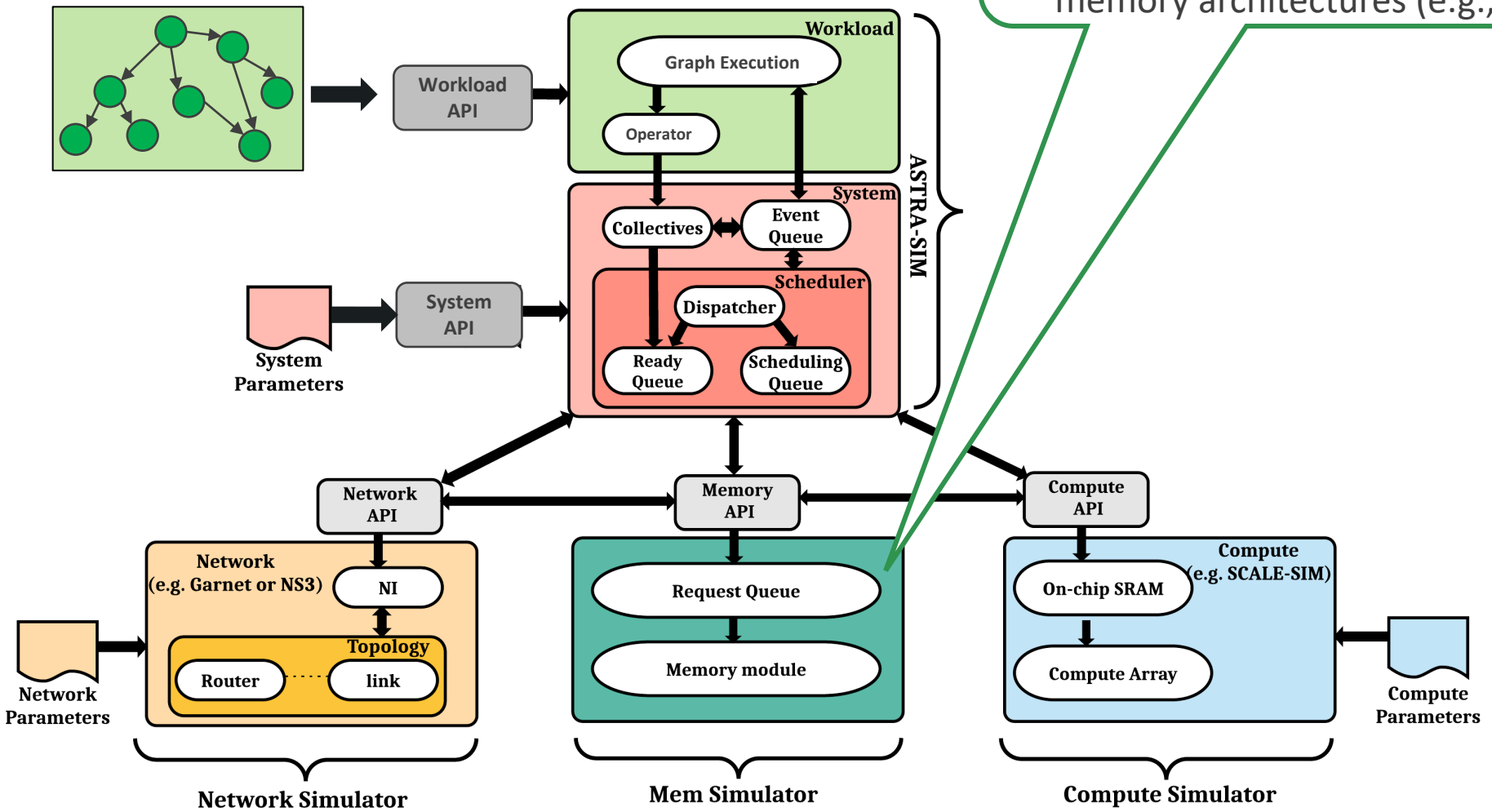
Model/Simulator	Purpose	Notable Feature
analytical	analytical equation-based simulation	fast simulation, hierarchical topologies
congestion-aware	congestion-aware analytical simulation	first-order congestion (queueing) modeling
Garnet	on-chip/scale-up network simulation	packetization, flow control, congestion
ns-3	inter-network simulation	RDMA, congestion-control
Genie	Transmit packets through real network	Measured network performance



Additional *proprietary* network simulators implementing the NetworkAPI being used by AMD, Alibaba, HPE Labs, some startups ..

Memory Layer

- ✓ Model data movement between remote memory to local memory
- ✓ Study emerging disaggregated memory architectures (e.g., CXL)

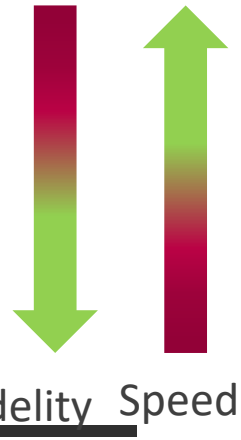


Memory Layer

- Simulates remote **memory behavior**
- Should support **multiple** memory architectures using RemoteMemoryAPI
  - Enabling the simulation of various scales/fidelity
- **Current version: only supports a simple analytical model**
  - Local vs Remote Mem BW
  - Pipelined Data Transfer

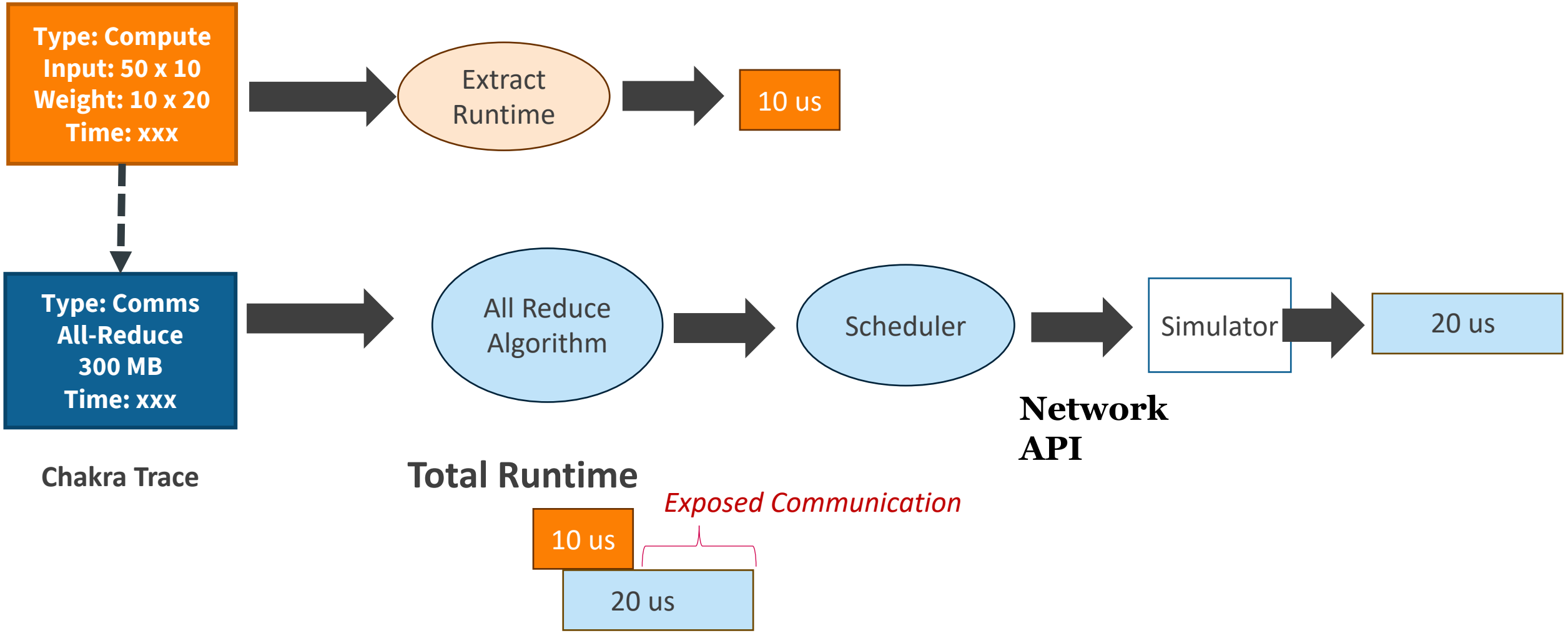
$$(Memory\ Access\ Time) = (Memory\ \ Access\ Latency) + (Tensor\ Size)/(Memory\ Bandwidth)$$

Model/Simulator	Purpose	Notable Feature
analytical	analytical equation-based simulation	fast simulation, disaggregated topologies



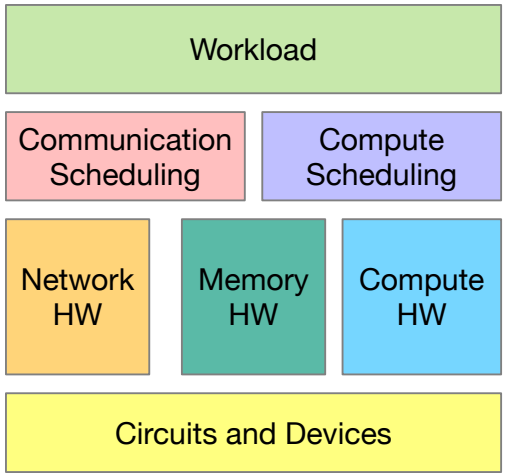
Example of ASTRA-sim in action

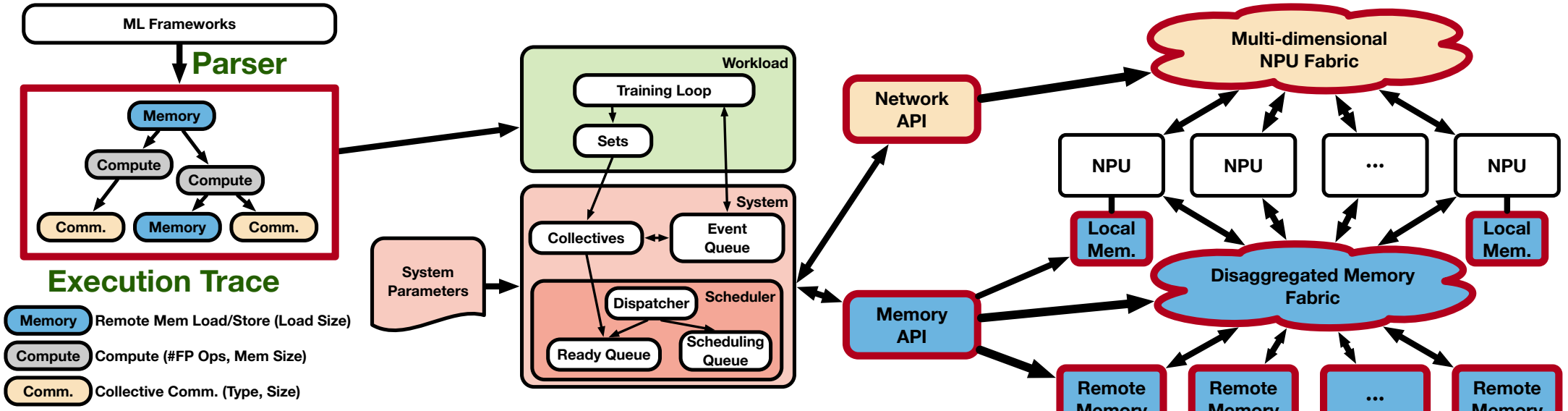
Simulating a system with new network fabric but same GPU



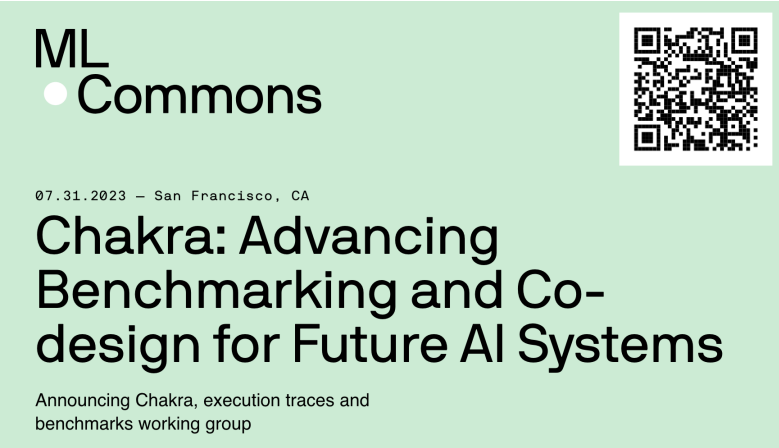
Summary of ASTRA-sim

- **Key Features**
  - Workload, System, Network, Memory, and Compute Layers
    - Plug-and-play support via APIs
  - Supports **arbitrary workload and parallelization** strategies
  - Models **system-level** behaviors
  - Supports **large-scale, multi-dimensional network** simulations
  - Runs various **compute simulation** backends

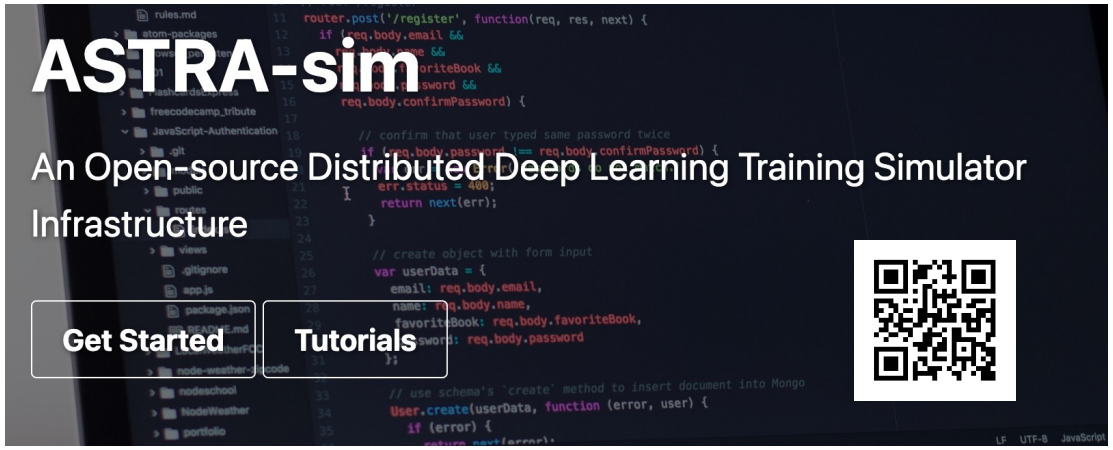




External Impact



- Chakra has been adopted by MLCommons!
  - maintainer of MLPerf
- I co-chair a working group with the following goals
  - Trace Format Standardization
  - Trace Collection support (PyTorch/TensorFlow)
  - Trace Replay and Simulation
  - Trace Benchmark Suite Creation



Tutorials

- ASPLOS (2022, 2023), ISCA (2022), MLSys (2022), MICRO 2024

Userbase (540 stars on github, 190 forks)

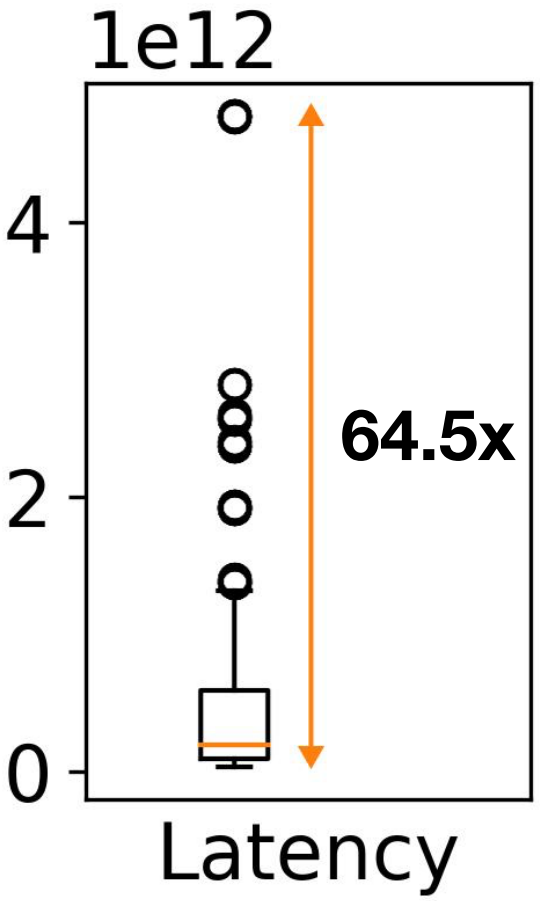
- Industry
  - AMD, Intel, NVIDIA, HPE Labs, Marvell, Alibaba, Keysight,
  - Many startups
- Open Compute Project (OCP) WG on Co-Design
- Many universities

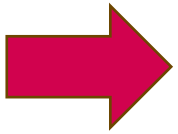
Outline

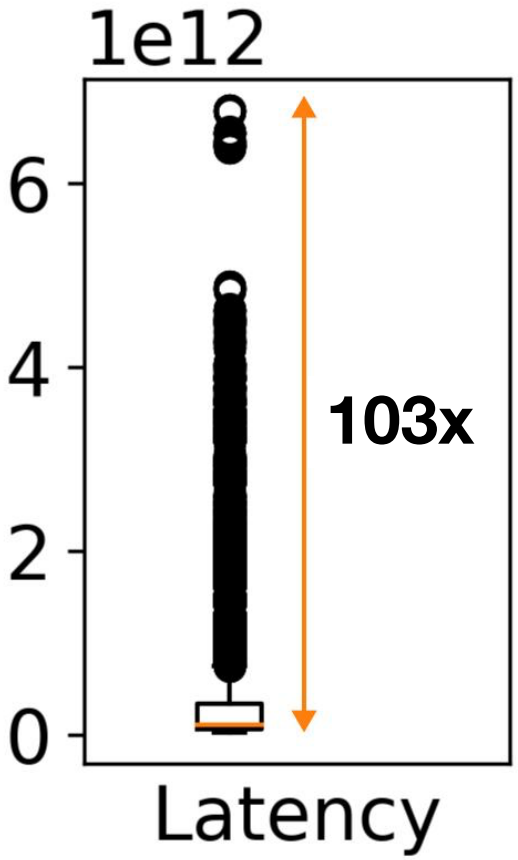
- Design Space of AI Platforms
- ASTRA-sim Ecosystem
- **Case Study: Using AI to Navigate Search Space**
- Conclusion

*Aditi Raju, Jared Ni, William Won, Changhai Man, Srivatsan Krishnan, Srinivas Sridharan, Amir Yazdanbakhsh, Tushar Krishna, Vijay Janapa Reddi, "**COSMIC: Enabling Full-Stack Co-Design and Optimization of Distributed Machine Learning Systems**", https://arxiv.org/abs/2505.15020*

Motivation







latency spread for training GPT3-175B just varying the workload parameters

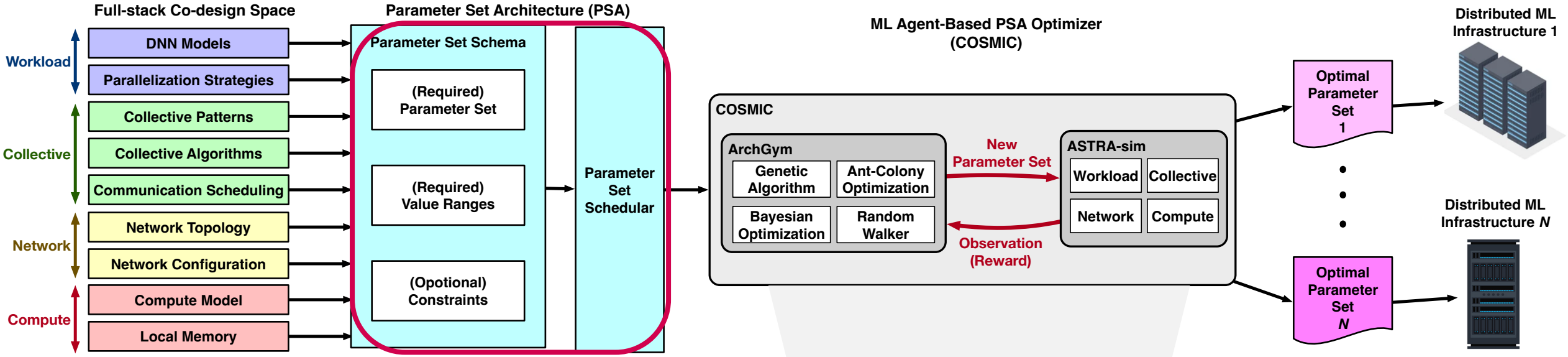
Full-stack co-design

Design-space Size

Workload Knob	Value Range	#Points
DP	{1, 2, 4, 8, ···, 512, 1024}	.
PP	{1, 2, 4, 8, ···, 512, 1024}	.
SP	{1, 2, 4, 8, ···, 512, 1024}	286
Weight Sharded	{0, 1}	2
Collective Knob	**Value Range**	**#Points**
Scheduling Policy	{LIFO, FIFO}	2
Collective Algorithm	MultiDim {Ring, Direct, RHD, DBT}	256
Chunks per Collective	{1, 2, 3, 4, ···, 32}	32
Multi-dim Collective	{Baseline, BlueConnect}	2
Network Knob	**Value Range**	**#Points**
Topology	MultiDim {Ring, Switch, FC}	81
NPUs per Dim	MultiDim {4, 8, 16}	81
Bandwidth per Dim	MultiDim {100, 200, 300, 400, 500}	625
Total #Points		7.69×10^{13}
Constraints		
product (DP, SP, PP) ≤ (Number of NPUs) = 1,024		
product (NPUs per Dim) = (Number of NPUs) = 1,024		

Even at 1 millisecond per configuration, exhaustive search would take ~2,500 years

Introducing COSMIC



Design Principles

- ✓ Separation of Concerns (AI Agents vs System Design Space)
- ✓ Automated Configuration of ML Design-Space
- ✓ Flexible Expression of System Design-Space

Evaluation Setup: AI System Design Space

System configurations used for evaluation

Knobs	System 1	System 2	System 3
Collective Knob			
Collective Algorithm	[RI, RI, RI, RHD]	[RI, DI, RI, RHD]	[DI, RHD, RI, RI]
Network Knob			
Topology	[RI, RI, RI, SW]	[RI, FC, RI, SW]	[FC, SW, RI, RI]
NPUs per Dim	[4, 4, 4, 8]	[4, 8, 4, 8]	[8, 16, 4, 4]
Bandwidth per Dim	[200, 200, 200, 50]	[375, 175, 150, 100]	[900, 100, 50, 12.5]
Compute Knob			
Compute Performance	459	10	900
Local Mem BW	2765	50	3000
	TPU-like	**4D Hier.**	**NV-like**

Target Workloads

Parameters	GPT3-175B	GPT3-13B	ViT-Base	ViT-Large
Number Layers*	96	40	12	24
Embedding Dimension	12288	5140	768	1024
FFN Dimension	49152	20560	3072	4096
Sequence Length	2048	2048	256	256
Number Heads	96	40	12	16

Parameter Set Architecture

Workload Knob	Value Range
DP	{1, 2, 4, 8, · · · , 1024, 2048}
PP	{1, 2, 4}
SP	{1, 2, 4, 8, · · · , 1024, 2048}
Weight Sharded	{0, 1}
Collective Knob	**Value Range**
Scheduling Policy	{LIFO, FIFO}
Collective Algorithm	MultiDim {Ring, Direct, RHD, DBT}
Chunks per Collective	{2, 4, 8, 16}
Multi-dim Collective	{Baseline, BlueConnect}
Network Knob	***Value Range***
Topology	MultiDim {Ring, Switch, FC}
NPUs per Dim	MultiDim {4, 8, 16}
Bandwidth per Dim	MultiDim {50, 100, 150, · · · , 450, 500}
Constraints	
product (DP, SP, PP) ≤ (Number of NPUs)	
product (NPUs per Dim) = (Number of NPUs)	

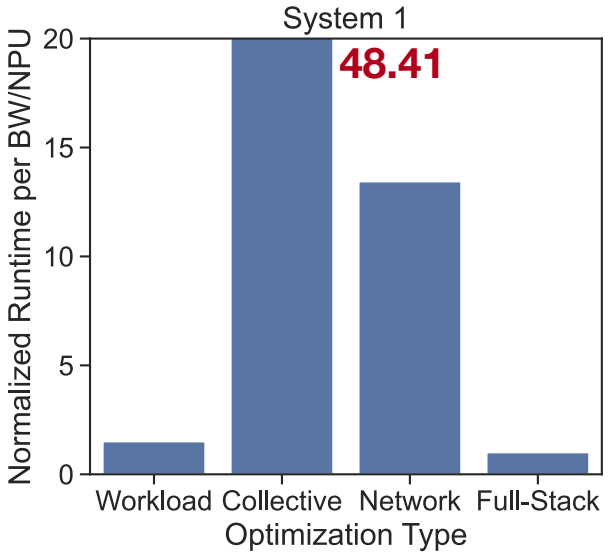
Evaluation Setup: ML Agents

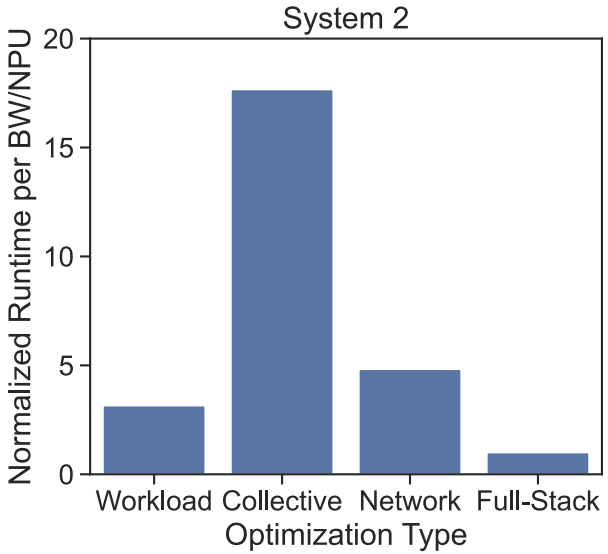
- Methods
  - RandomWalker (RW)
  - Genetic Algorithm (GA)
  - Ant Colony Optimization (ACO)
  - Bayesian Optimization (BO)

- Optimization Objectives
  - Runtime per BW/NPU
  - Runtime per Network Cost.

Results

Performance/Bandwidth



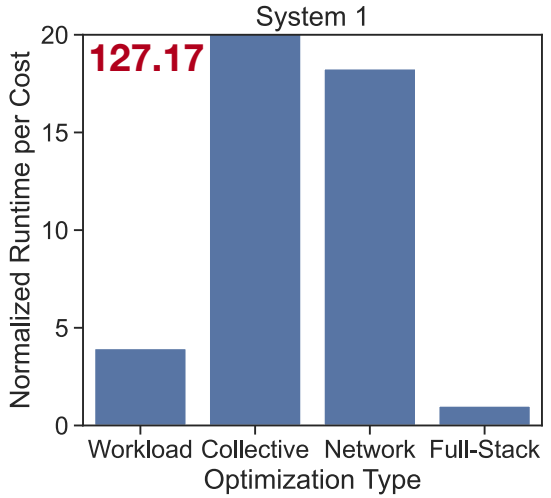


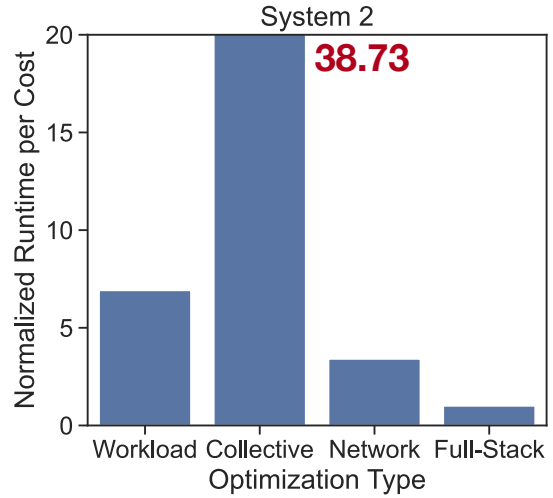
1.50 to 48.41x

3.15 to 7.67x

Highest Gains from Workload-opt

Performance/Cost



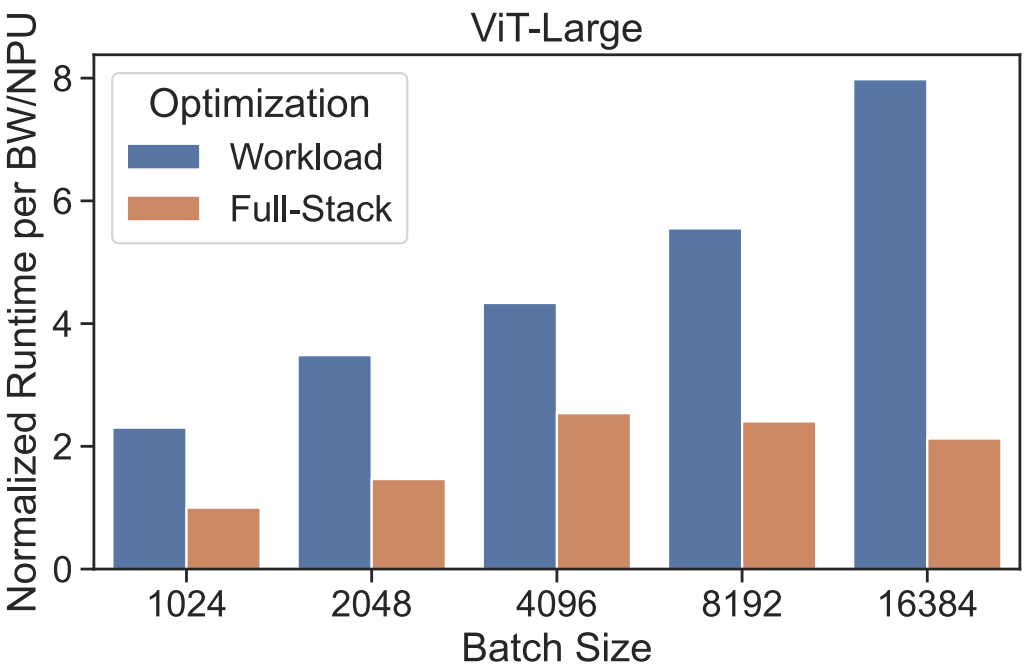


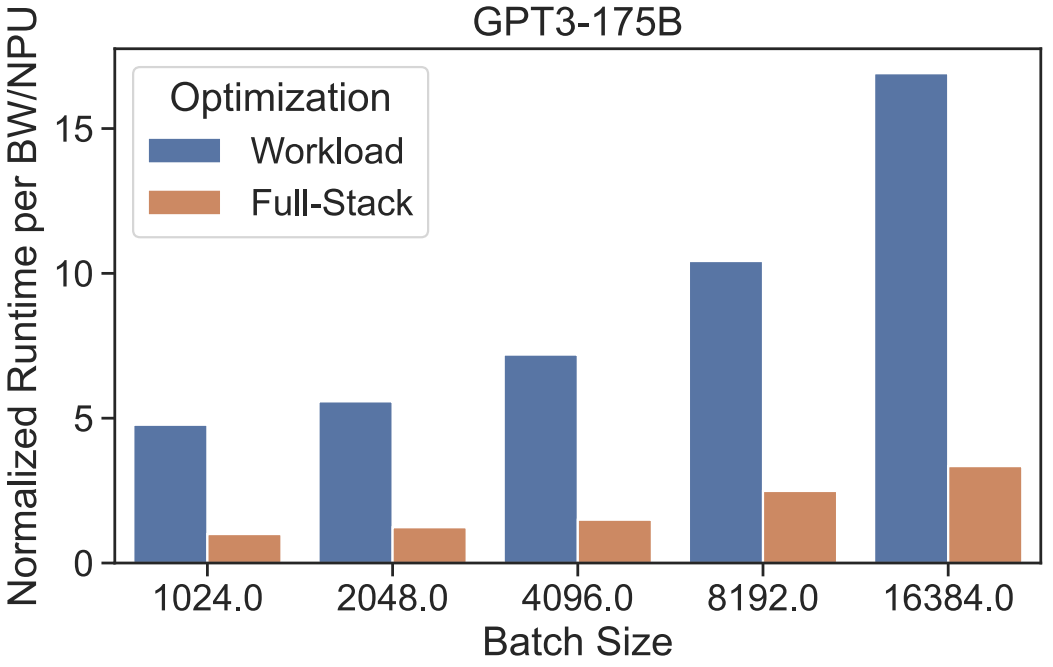
3.94- to 127.17x

3.40 – 38.73x

Cost-optimization can lead to diverse choices

Scalability





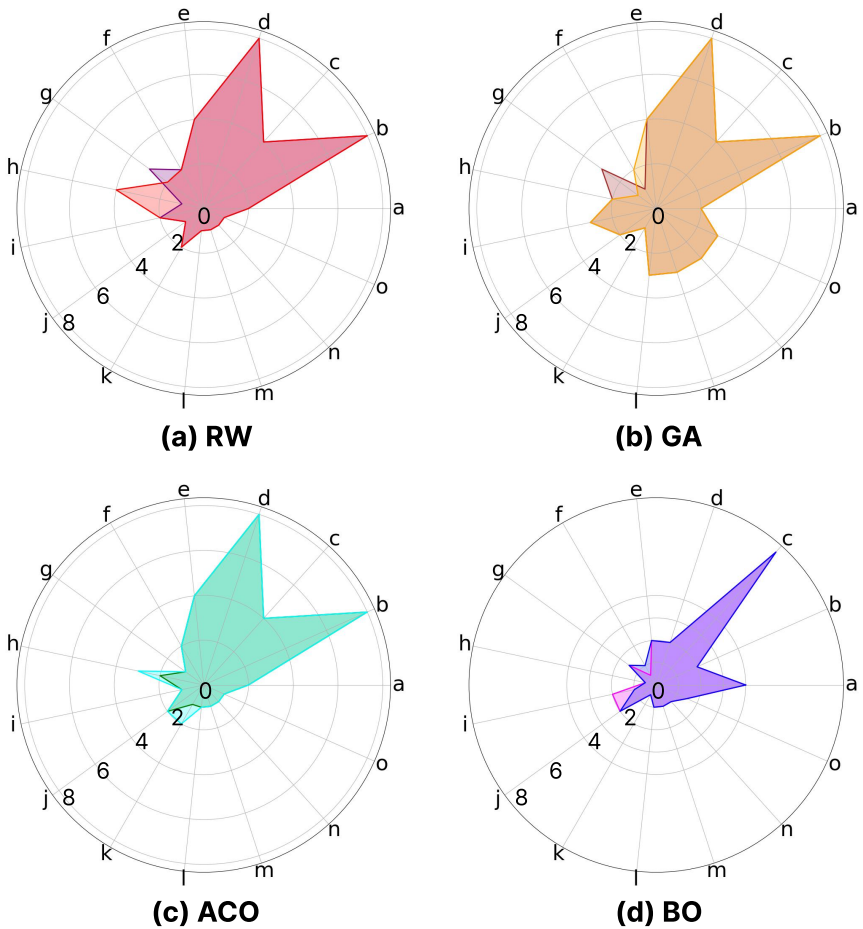
Value of Full-stack Opt increases with System Size

Designs Found for Diff Use Cases

*: **Searching**: Searched parameter; **Fixed**: Fixed value;

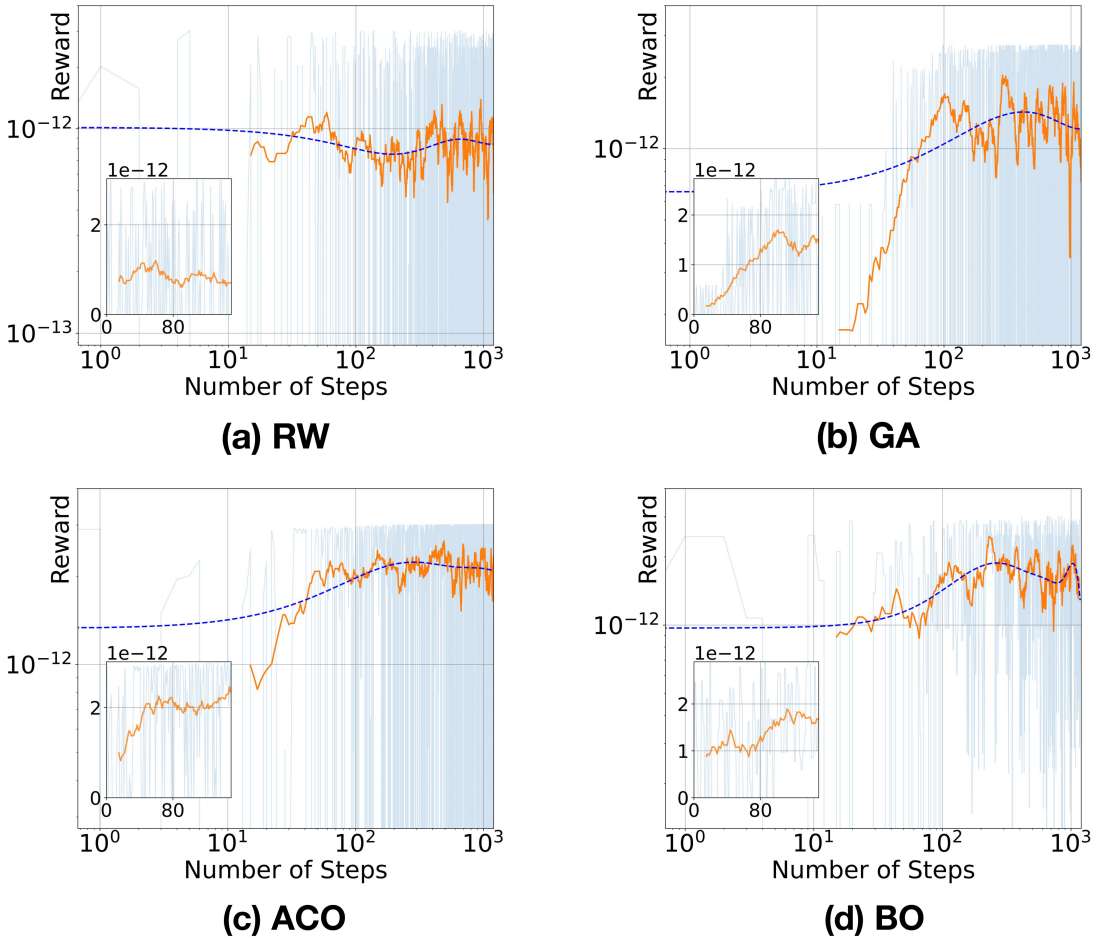
	Expr. 1	Expr. 2.1	Expr. 2.2
Observations	**Multi-Model**	**Chat Inference**	**QA Inference**
Network Knobs			
Topology	[RI, FC, RI, FC]	[RI, RI, RI, RI]	[RI, RI, RI, RI]
NPUs-count	[16, 4, 4, 4]	[4, 16, 4, 4]	[16, 4, 4, 4]
Bandwidth per Link	[50, 50, 50, 50]	[50, 50, 50, 50]	[50, 50, 50, 50]
Collective Knobs			
Scheduling Policy	LIFO	LIFO	LIFO
Chunks per Collective	4	2	2
Collective Algorithm	[RI, RI, RI, RI]	[RHD, DBT, DBT, DI]	[DI, DI, DBT, RHD]
Multi-dim Collective	Baseline	BlueConnect	Baseline
Workload Knobs			
Number of NPUs	1024	1024	1024
DP, PP, SP, TP	2, 1, 8, 64	8, 4, 8, 4	8, 4, 8, 4
Weight Sharded	1	1	1

Comparing ML Agents



Consistency in key performance-critical parameters, with variance in less impactful parameters

The RW agent does not leverage history, resulting in a relatively flat reward curve

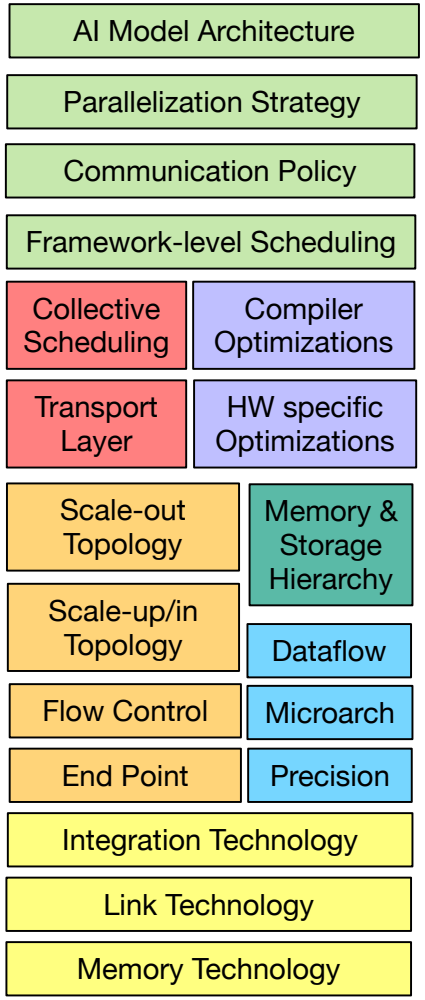


Other agents exhibit learning behavior

Outline

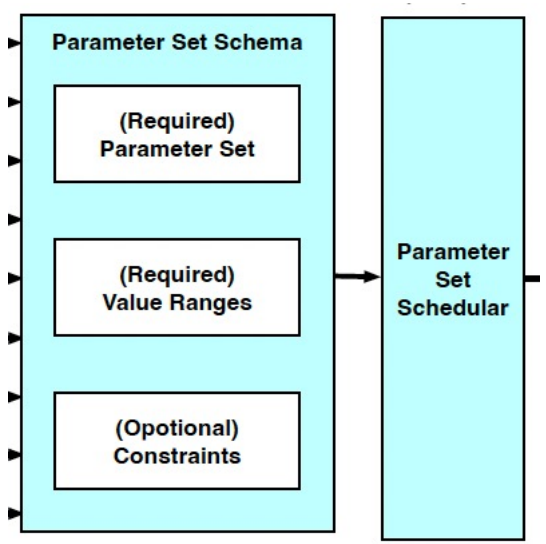
- Design Space of AI Platforms
- ASTRA-sim Ecosystem
- Case Study: Using AI to Navigate Search Space
- **Conclusion**

Conclusions and Takeaways

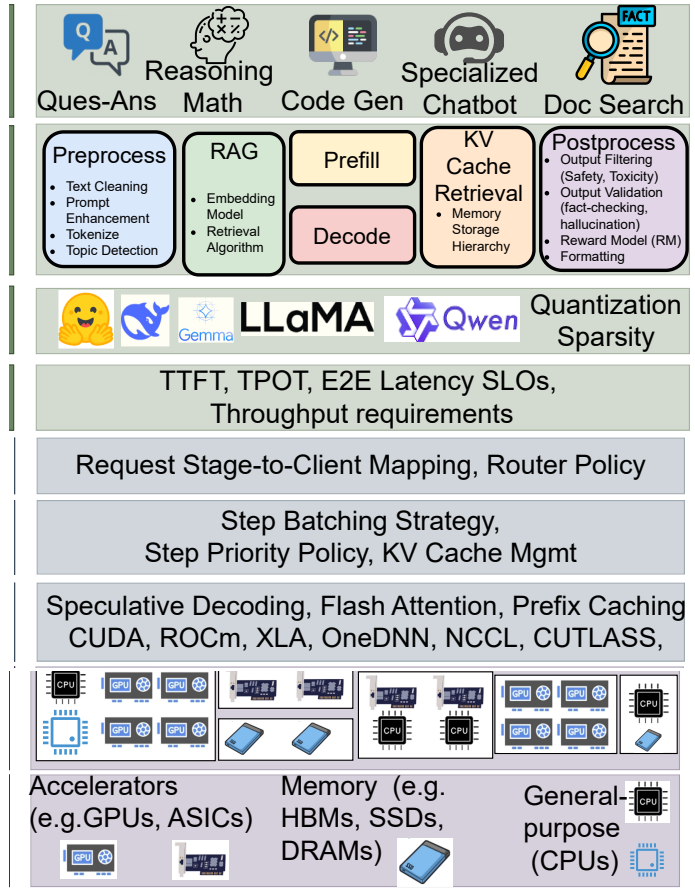


Complex Design-Space of Distributed AI Training Platforms

Framework for Multi-fidelity design-space Exploration



Using AI Agents to Search via Parameter Set Architecture



Design Spaces are becoming more complex (esp with AI Inference)!

Thank you!