SAMSUNG AI Cambridge

Beyond Status-Quo NPUs: Hardware-Native Multi-Tenancy & Adaptive Inference

Stylianos I. Venieris

- Sept 10th 2024
- Samsung AI Center, Cambridge, UK

Distributed AI Group @ Samsung AI

SAMSUNG AI Cambridge

Stylianos (Stelios) Venieris

Rui Li

Alexandros Kouris

Hongxiang Fan

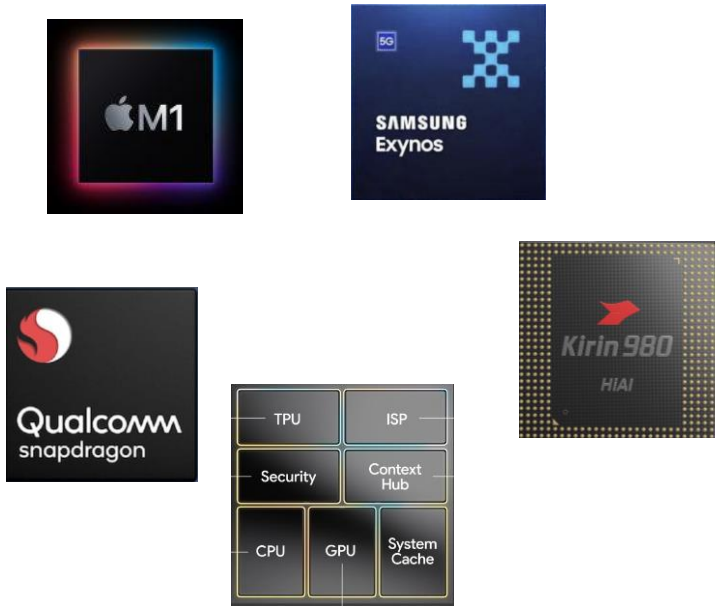
Konstantin Mishchenko

Hao (Mark) Chen

The Power of Neural Processing Units (NPUs)

SAMSUNG AI Center - Cambridge

- Becoming ubiquitous across mobile devices, robots and cloud servers
- Backbone of deployed AI, enabling everything from facial recognition to advanced chatbots



The Power of Neural Processing Units (NPUs)

SAMSUNG AI Center - Cambridge

- Becoming ubiquitous across mobile devices, robots and cloud servers
- Backbone of deployed AI, enabling everything from facial recognition to advanced chatbots

The Power of Neural Processing Units (NPUs)

SAMSUNG AI Center - Cambridge

- Becoming ubiquitous across mobile devices, robots and cloud servers
- Backbone of deployed AI, enabling everything from facial recognition to advanced chatbots

Existing NPUs are optimised for:

- CNNs | Single-DNN execution

New Challenges

1. New DNN architectures (Transformers, Diffusion)
2. Multi-tenancy (multi-DNN workloads)
3. Diverse KPIs (latency, throughput, accuracy)

The Power of Neural Processing Units (NPUs)

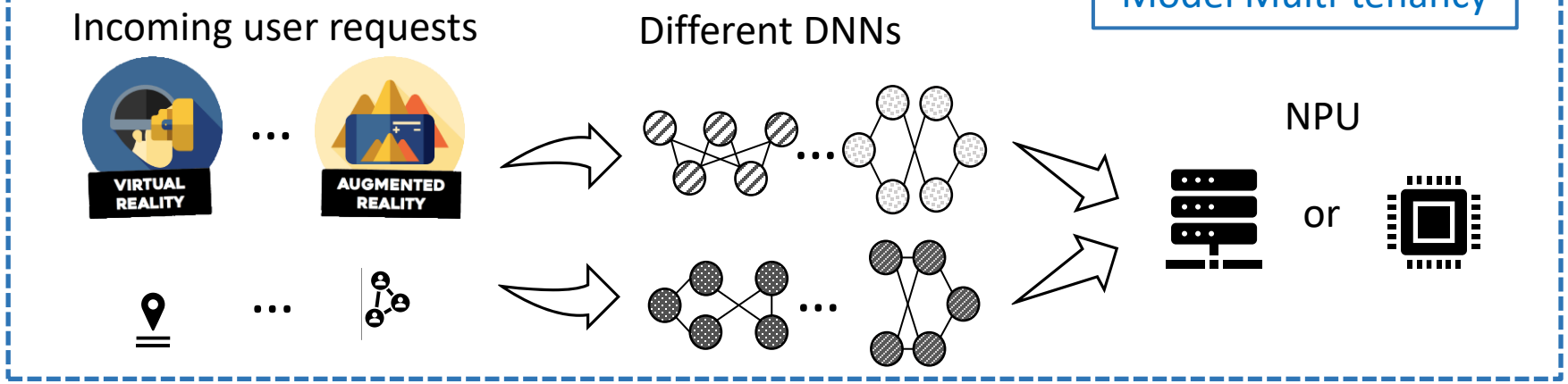
SAMSUNG AI Center - Cambridge

Existing NPUs are optimised for:

- CNNs | Single-DNN execution

New Challenges

1. New DNN architectures (Transformers, Diffusion)
2. Multi-tenancy (multi-DNN workloads)
3. Diverse KPIs (latency, throughput, accuracy)



A New Class of NPUs: Multi-DNN Accelerators

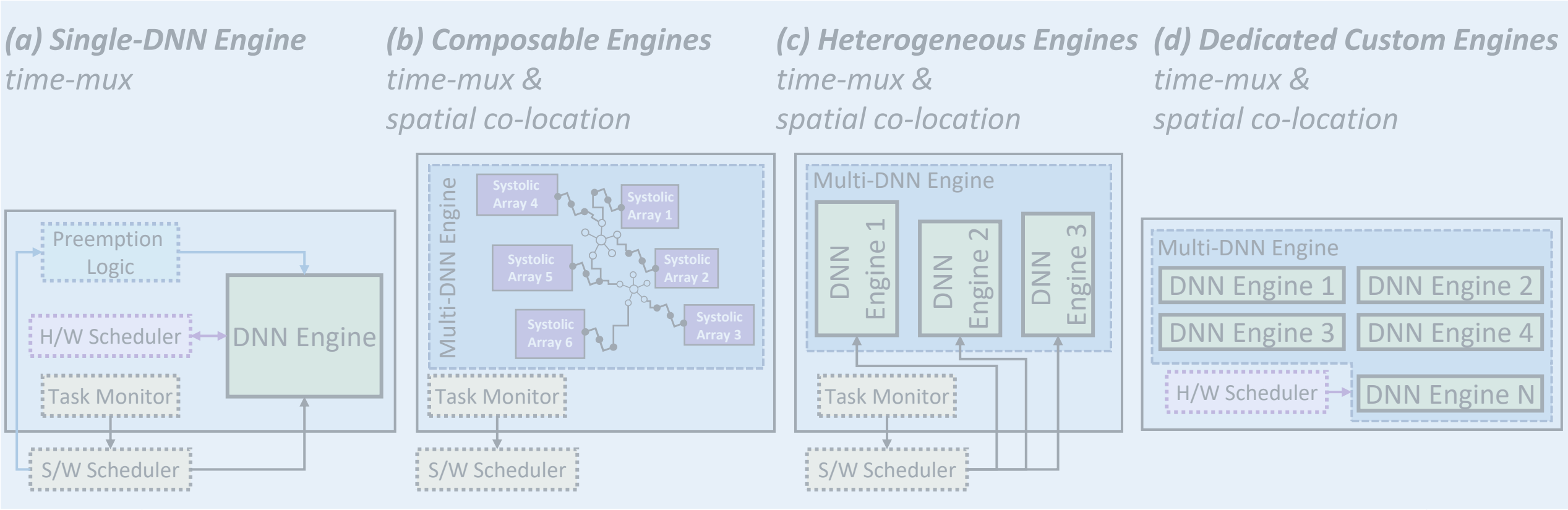
SAMSUNG AI Center - Cambridge

- NPUs with native support for multi-model execution
- Key design decisions
  - Inter-DNN parallelisation strategy →
  - DNN Engine Architecture →
  - Scheduling →

Approach 1	**Approach 2**
Time-mux	Spatial Co-location
Enhancing existing architectures	Custom Designs
Static	Dynamic

Landscape of Multi-DNN Accelerators

SAMSUNG AI Center - Cambridge



Programmability

Customisation

S.I. Venieris, C.S. Bouganis, and N. D. Lane.
Multi-DNN Accelerators for Next-Generation AI Systems. *IEEE Computer,* 2023.

Landscape of Multi-DNN Accelerators

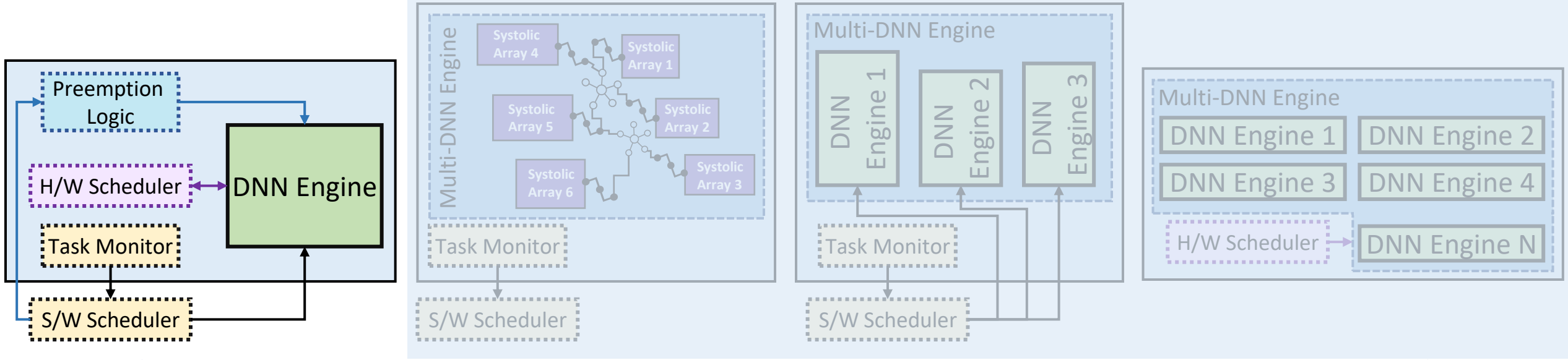
SAMSUNG AI Center - Cambridge

(a) Single-DNN Engine
time-mux

(b) Composable Engines
time-mux &
spatial co-location

(c) Heterogeneous Engines
time-mux &
spatial co-location

(d) Dedicated Custom Engines
time-mux &
spatial co-location



Programmability ⟷ Customisation

S.I. Venieris, C.S. Bouganis, and N. D. Lane.
Multi-DNN Accelerators for Next-Generation AI Systems. *IEEE Computer,* 2023.

Landscape of Multi-DNN Accelerators

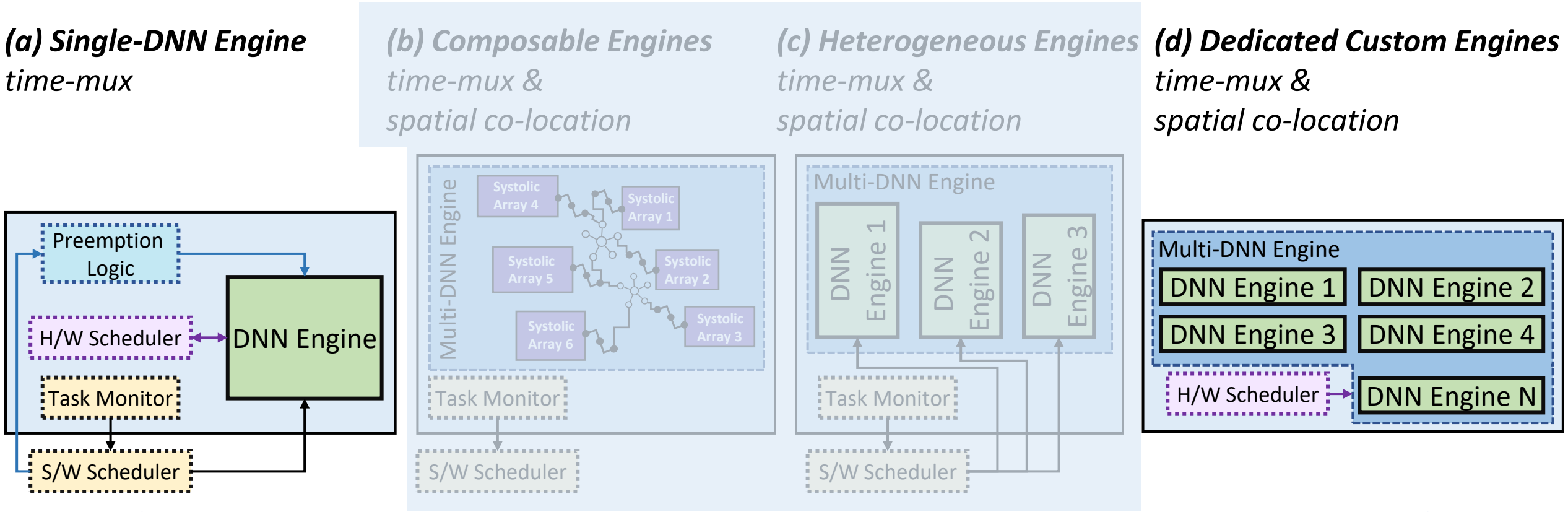
SAMSUNG AI Center - Cambridge

(a) Single-DNN Engine
time-mux

(b) Composable Engines
time-mux &
spatial co-location

(c) Heterogeneous Engines
time-mux &
spatial co-location

(d) Dedicated Custom Engines
time-mux &
spatial co-location



Programmability — Customisation

S.I. Venieris, C.S. Bouganis, and N. D. Lane.
Multi-DNN Accelerators for Next-Generation AI Systems. *IEEE Computer,* 2023.

Landscape of Multi-DNN Accelerators

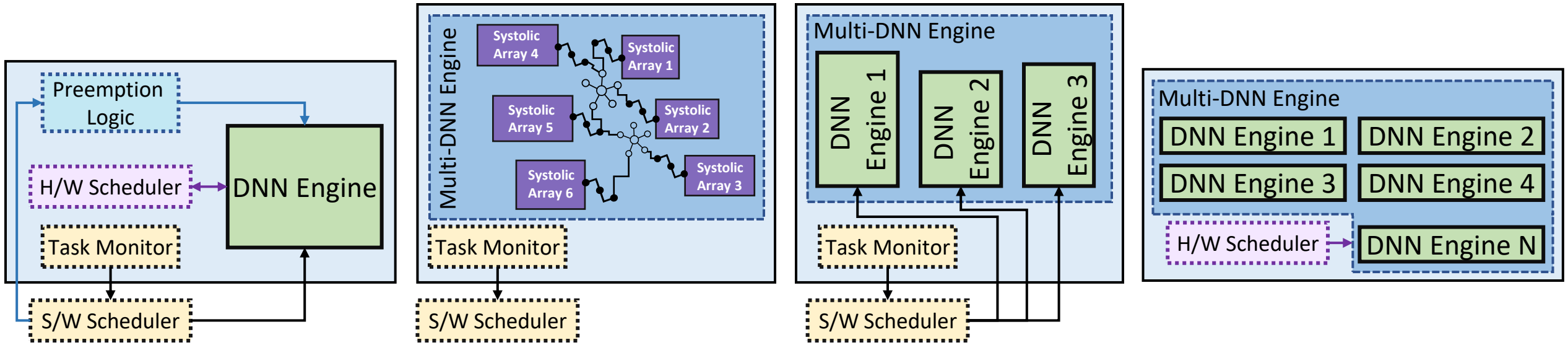
SAMSUNG AI Center - Cambridge

(a) Single-DNN Engine
time-mux

(b) Composable Engines
time-mux &
spatial co-location

(c) Heterogeneous Engines
time-mux &
spatial co-location

(d) Dedicated Custom Engines
time-mux &
spatial co-location



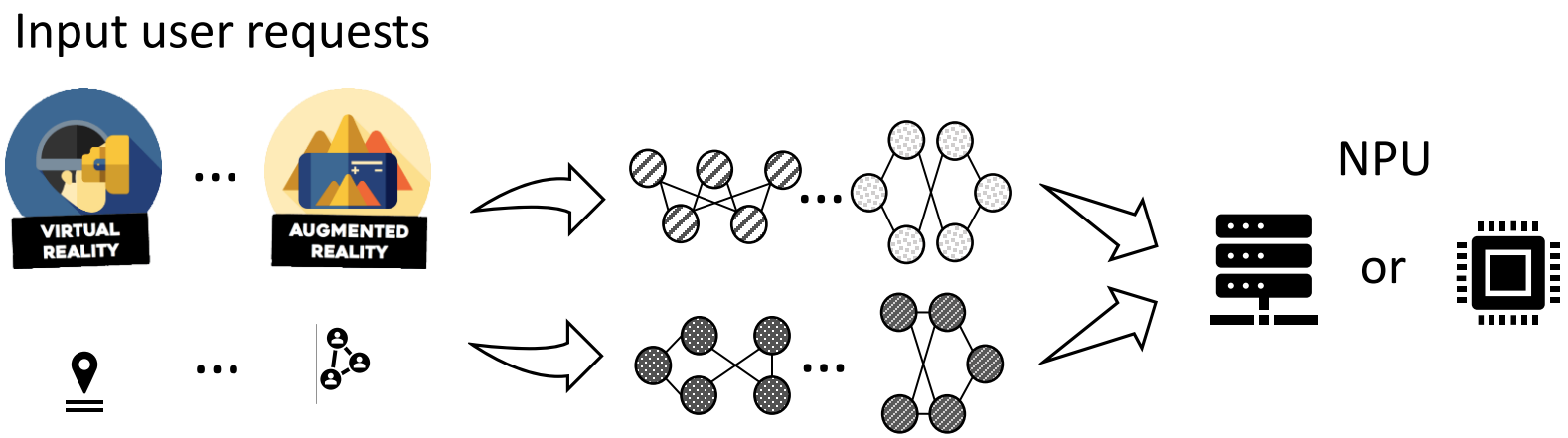
Programmability ⟷ Customisation

S.I. Venieris, C.S. Bouganis, and N. D. Lane.
Multi-DNN Accelerators for Next-Generation AI Systems. *IEEE Computer,* 2023.

Case Study: Sparse Multi-DNN Workloads

SAMSUNG AI Center - Cambridge

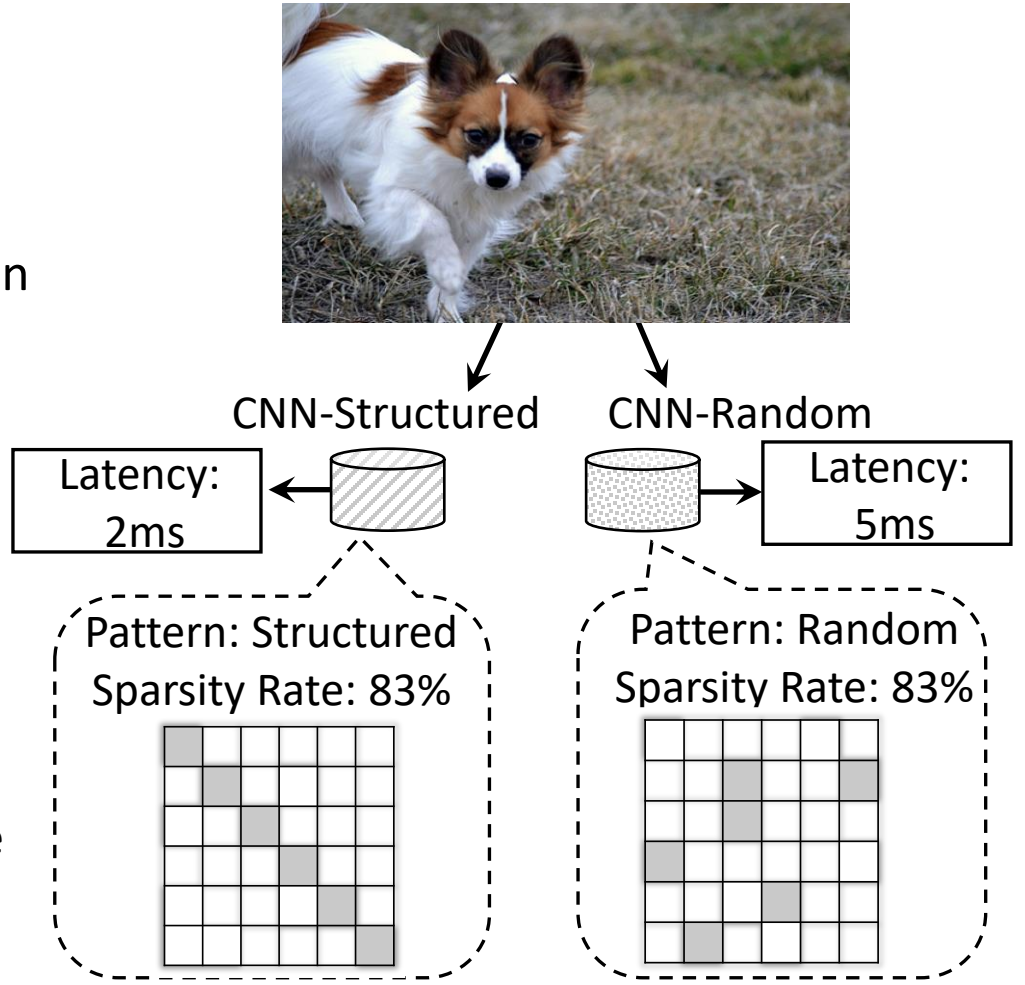
- Sparse multi-DNN workloads
  - Different sparsification methods, e.g. channel-wise, N:M block-wise, random
  - Diverse inference requests: <input, DNN, sparsification method, KPI>
- Key component: Scheduling
  - Determine the execution order of multi-DNNs
  - Focus on layer-wise execution on the NPU



Case Study: Sparse Multi-DNN Workloads

SAMSUNG AI Center - Cambridge

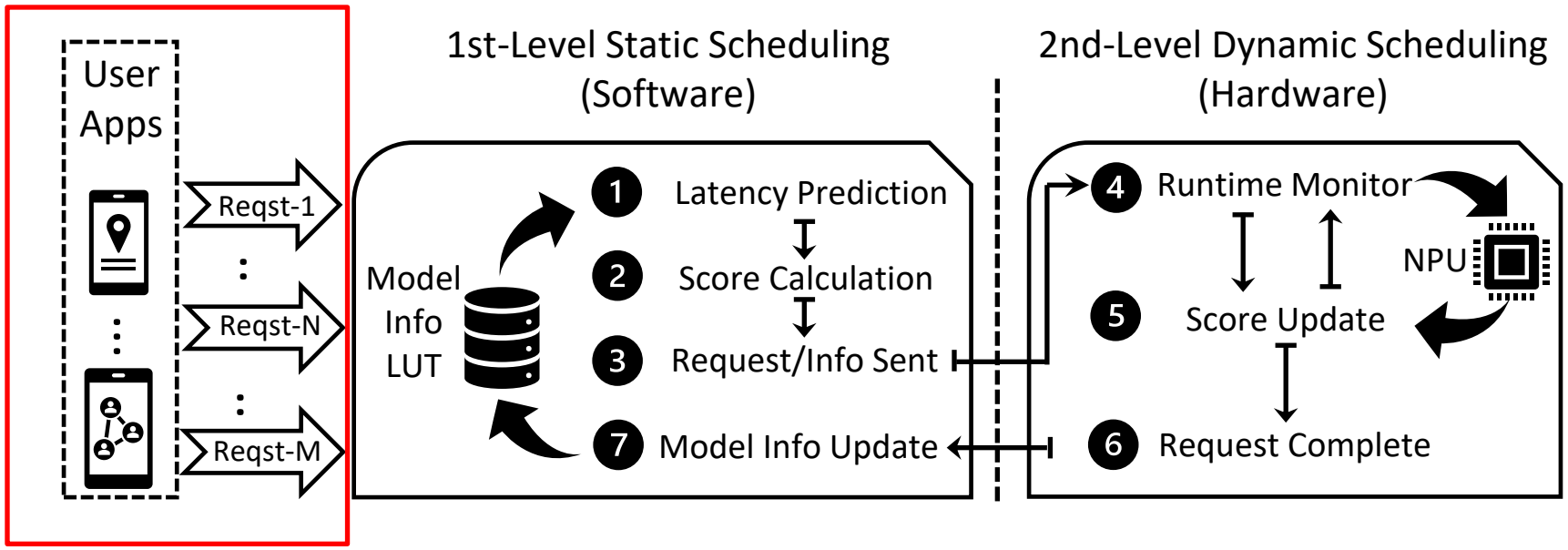
- Current approaches
  - Lack of adequate sparsity support
  - Only consider average sparsity rate
- Untapped opportunity: Fine-grained sparsity information
  - Sparsity pattern
  - Sparsity dynamicity
- Sparsity pattern → hardware efficiency variability
  - NVIDIA Tensor Core: N:M block-wise sparsity
  - Samsung Exynos NPU: Random sparsity
- Sparsity dynamicity → Runtime variance
  - Varied latency performance can affect the best schedule



Case Study: Dysta

SAMSUNG AI Center - Cambridge

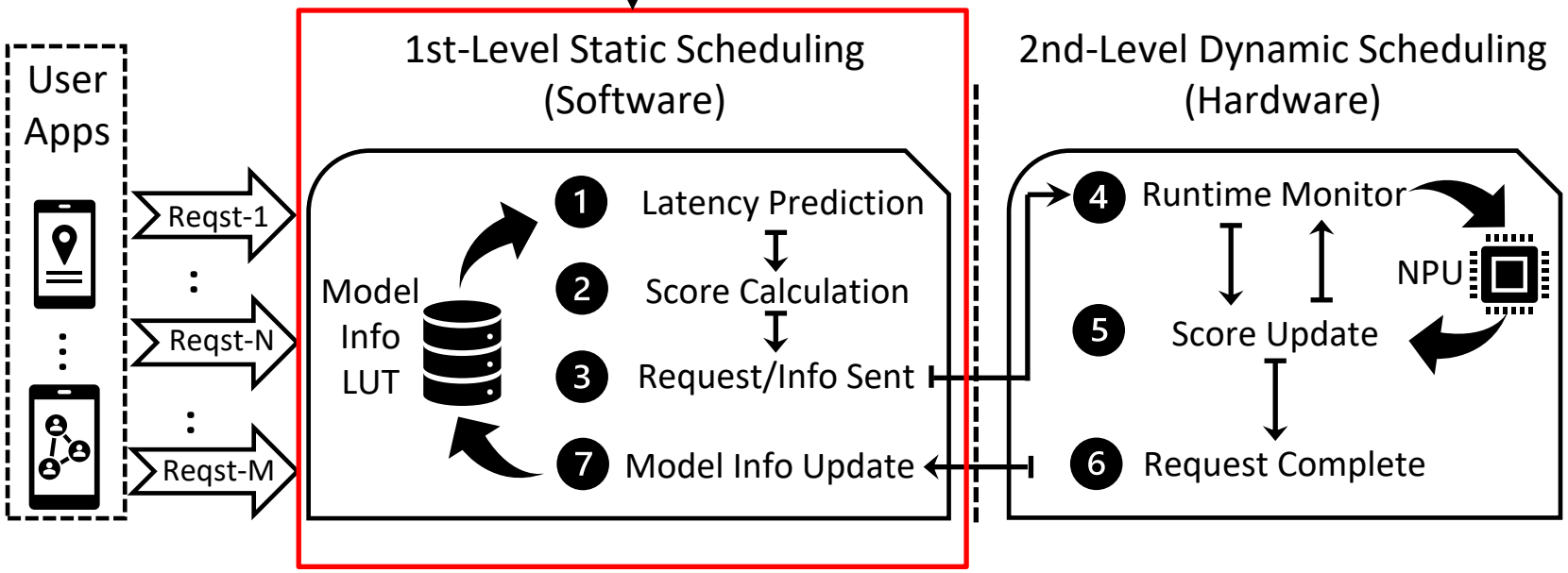
- **Problem:** Schedule variably sparse DNNs on zero-skipping NPUs, when each request has different <input, DNN, sparsity pattern, runtime sparsity level, KPI>
- **Approach:** Sparsity-aware bi-level scheduling
  - Capture both **sparsity pattern (static)** and **dynamic sparsity (run-time) information**
  - SW-based static scheduler: Initial schedule
  - HW-based dynamic scheduler: Run-time schedule refinement



Case Study: Dysta

SAMSUNG AI Center - Cambridge

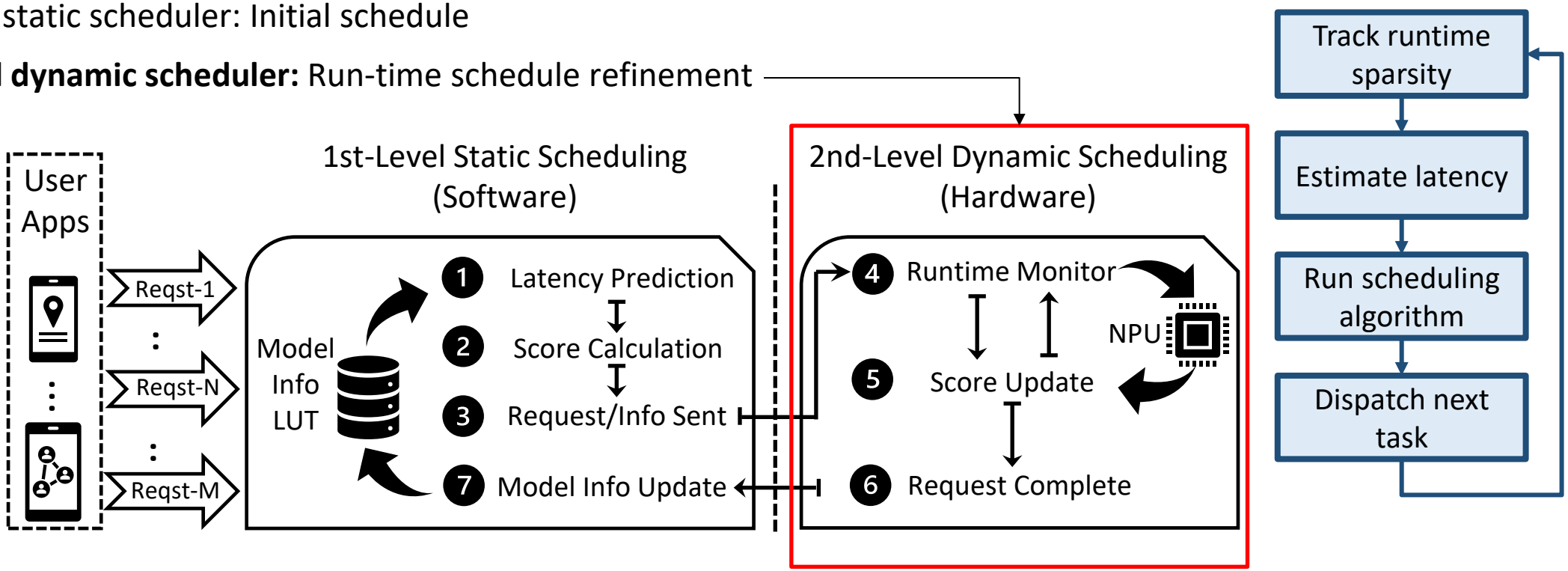
- **Problem:** Schedule variably sparse DNNs on zero-skipping NPUs, when each request has different <input, DNN, sparsity pattern, runtime sparsity level, KPI>
- **Approach:** Sparsity-aware bi-level scheduling
  - Capture both sparsity pattern (static) and dynamic sparsity level (run-time) information
  - **SW-based static scheduler:** Initial schedule
  - HW-based dynamic scheduler: Run-time schedule refinement



Case Study: Dysta

SAMSUNG AI Center - Cambridge

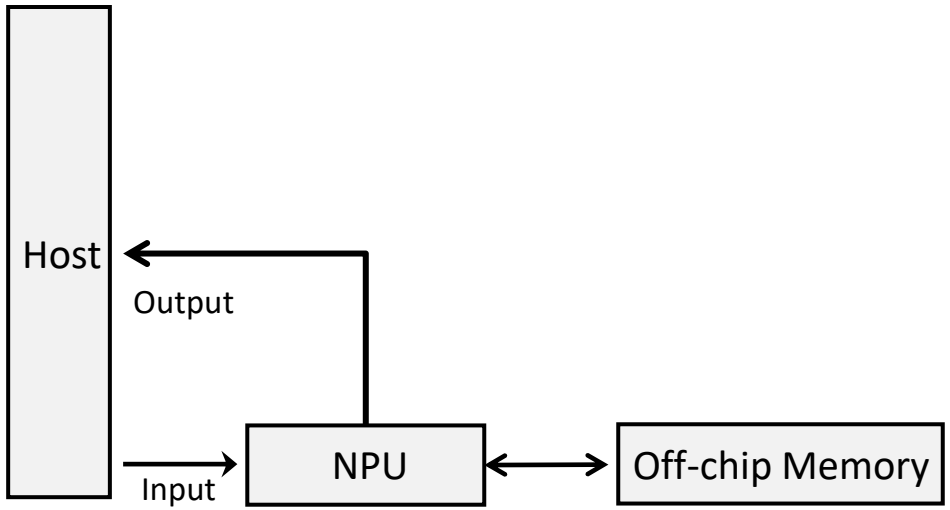
- **Problem:** Schedule variably sparse DNNs on zero-skipping NPUs, when each request has different <input, DNN, sparsity pattern, runtime sparsity level, KPI>
- **Approach:** Sparsity-aware bi-level scheduling
  - Capture both sparsity pattern (static) and dynamic sparsity level (run-time) information
  - SW-based static scheduler: Initial schedule
  - **HW-based dynamic scheduler:** Run-time schedule refinement



Case Study: Dysta – Enhancing existing NPUs

SAMSUNG AI Center - Cambridge

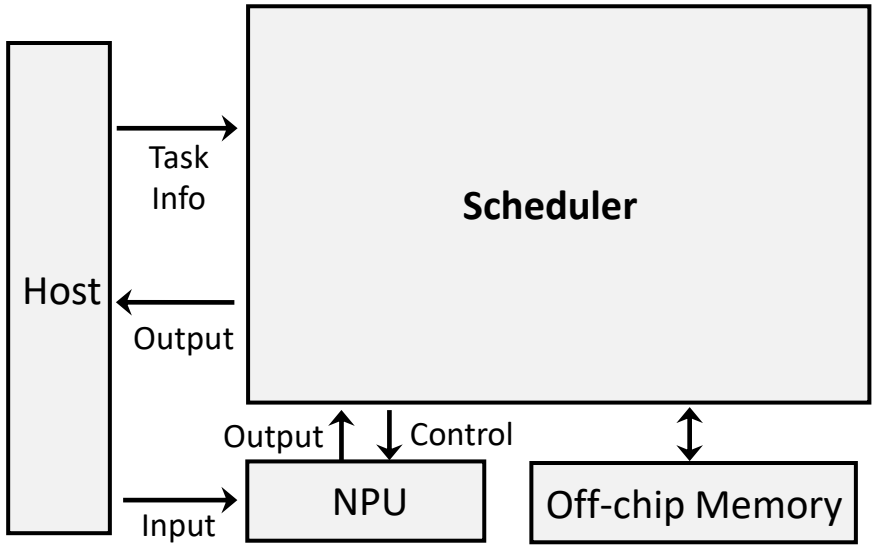
- Hardware scheduler
  - FIFOs: Track per-task information
  - Monitor: Track runtime sparsity
  - Compute unit: Estimate latency and update schedule
  - Reconfigurable design: Reuse resources for latency estimation and scheduling algorithm



Case Study: Dysta – Enhancing existing NPUs

SAMSUNG AI Center - Cambridge

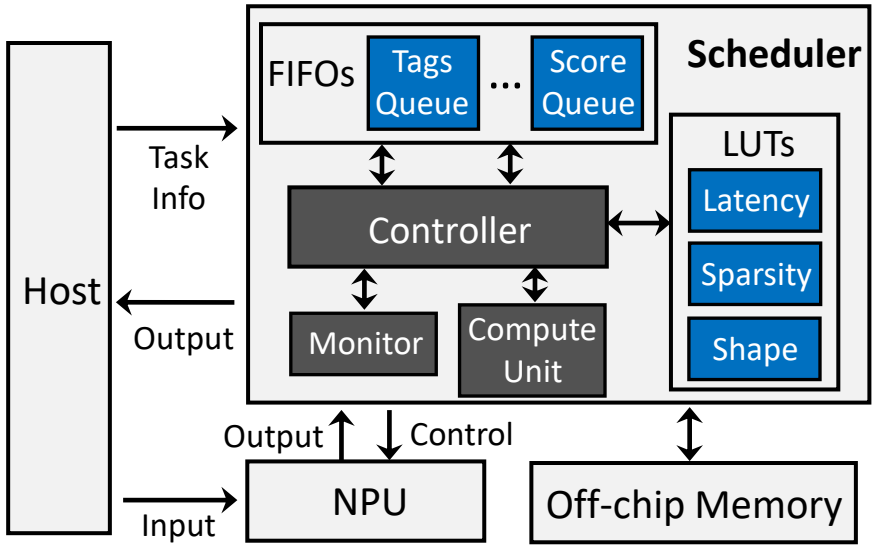
- Hardware scheduler
  - FIFOs: Track per-task information
  - Monitor: Track runtime sparsity
  - Compute unit: Estimate latency and update schedule
  - Reconfigurable design: Reuse resources for latency estimation and scheduling algorithm



Case Study: Dysta – Enhancing existing NPUs

SAMSUNG AI Center - Cambridge

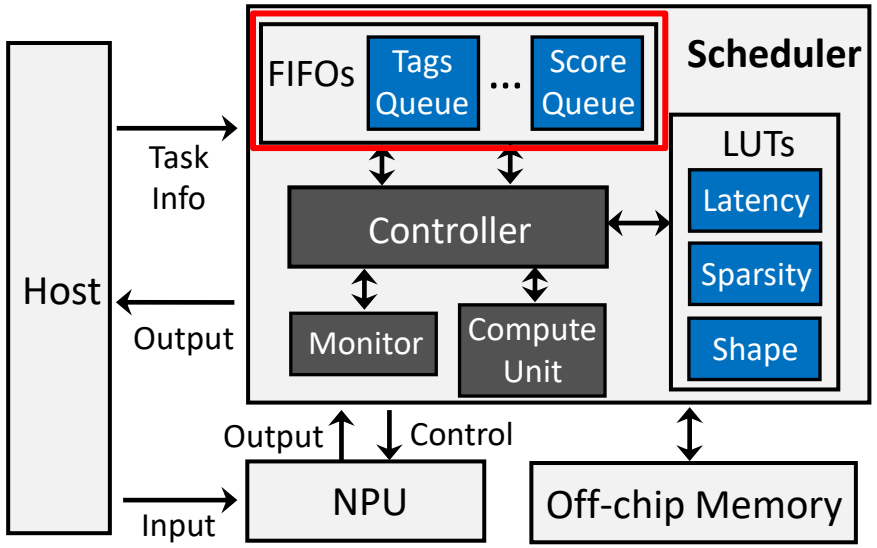
- Hardware scheduler
  - FIFOs: Track per-task information
  - Monitor: Track runtime sparsity
  - Compute unit: Estimate latency and update schedule
  - Reconfigurable design: Reuse resources for latency estimation and scheduling algorithm



Case Study: Dysta – Enhancing existing NPUs

SAMSUNG AI Center - Cambridge

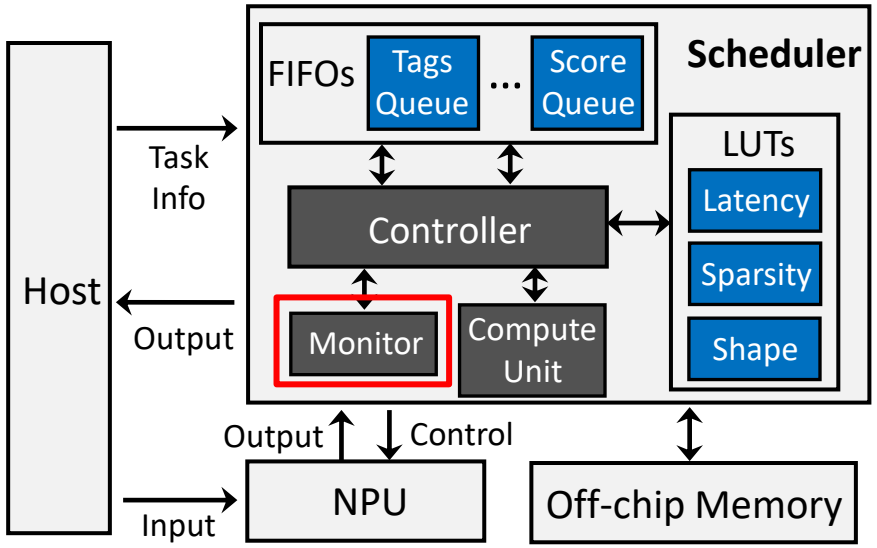
- Hardware scheduler
  - FIFOs: Track per-task information
  - Monitor: Track runtime sparsity
  - Compute unit: Estimate latency and update schedule
  - Reconfigurable design: Reuse resources for latency estimation and scheduling algorithm

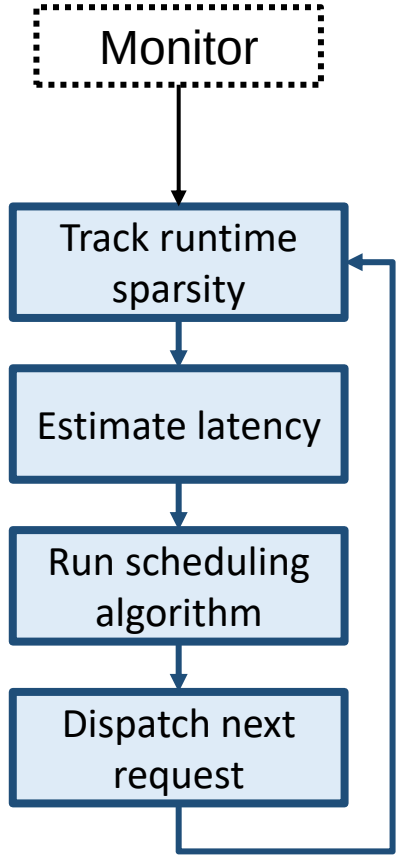


Case Study: Dysta – Enhancing existing NPUs

SAMSUNG AI Center - Cambridge

- Hardware scheduler
  - FIFOs: Track per-task information
  - Monitor: Track runtime sparsity
  - Compute unit: Estimate latency and update schedule
  - Reconfigurable design: Reuse resources for latency estimation and scheduling algorithm

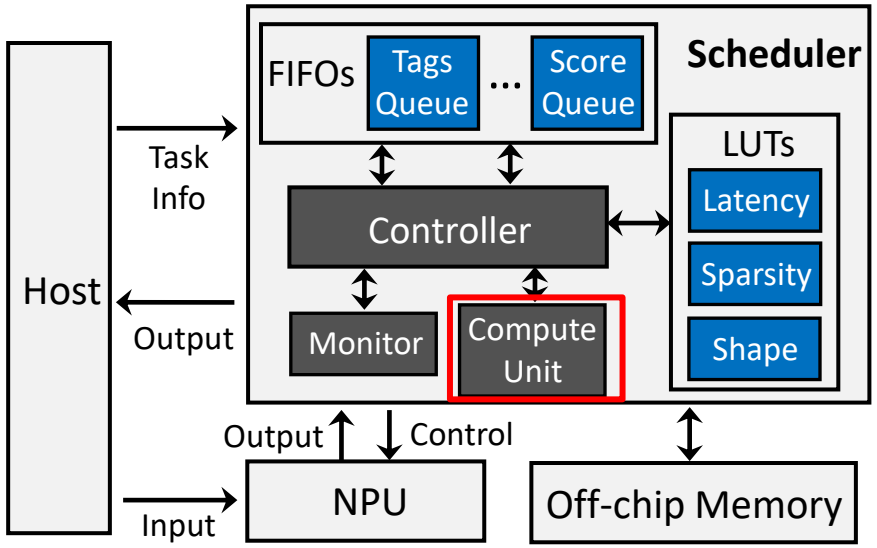


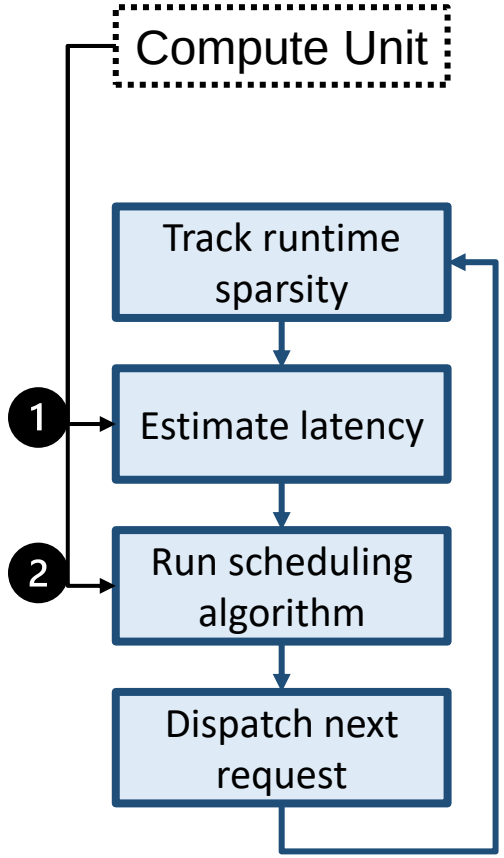


Case Study: Dysta – Enhancing existing NPUs

SAMSUNG AI Center - Cambridge

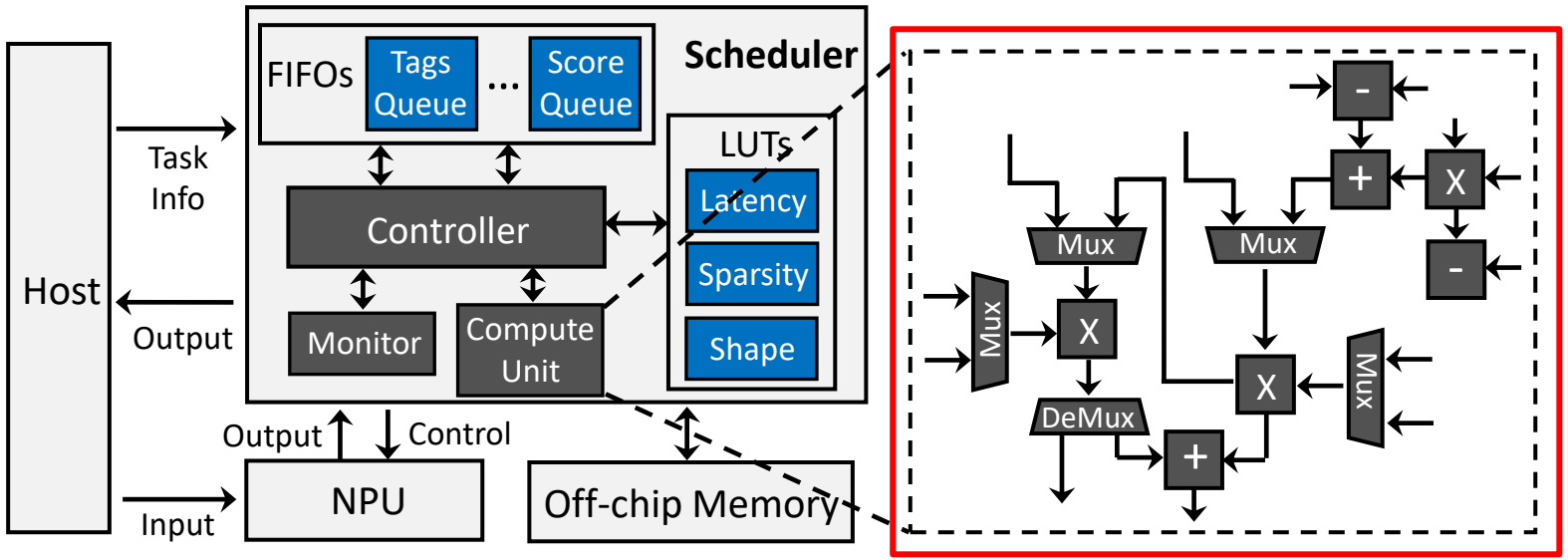
- Hardware scheduler
  - FIFOs: Track per-task information
  - Monitor: Track runtime sparsity
  - Compute unit: Estimate latency and update schedule
  - Reconfigurable design: Reuse resources for latency estimation and scheduling algorithm

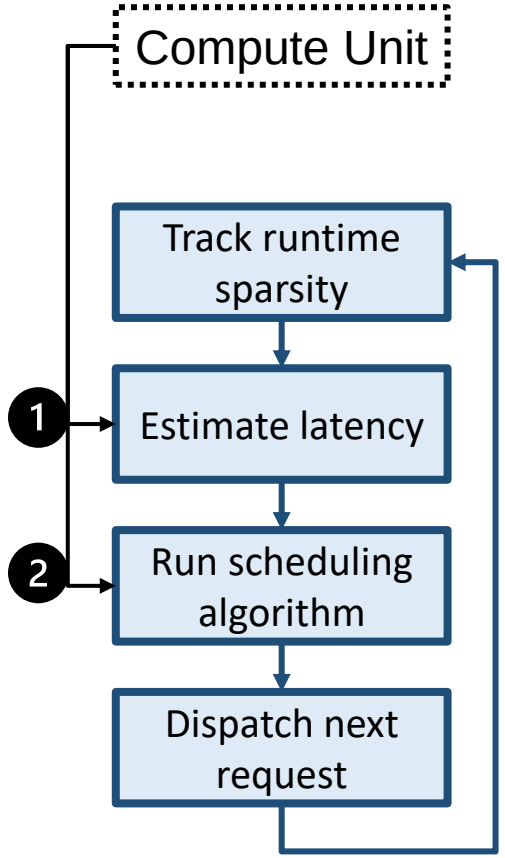




Case Study: Dysta – Enhancing existing NPUs

- Hardware scheduler
  - FIFOs: Track per-task information
  - Monitor: Track runtime sparsity
  - Compute unit: Estimate latency and update schedule
  - Reconfigurable design: Reuse resources for latency estimation and scheduling algorithm

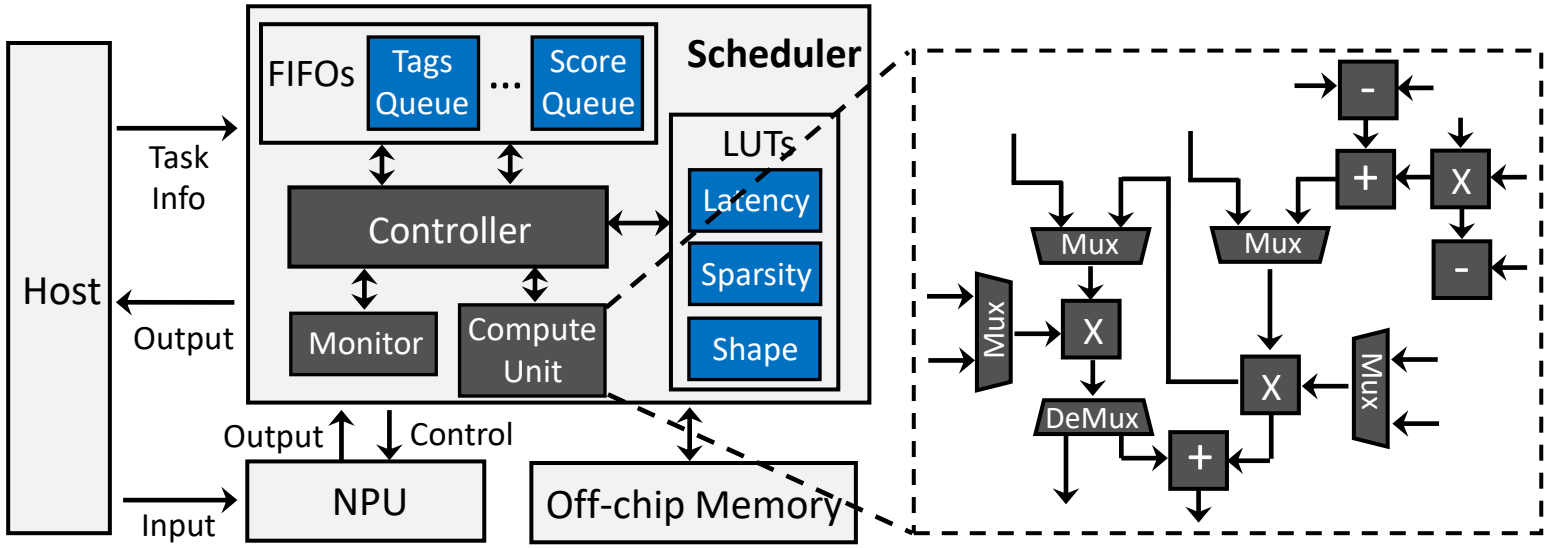


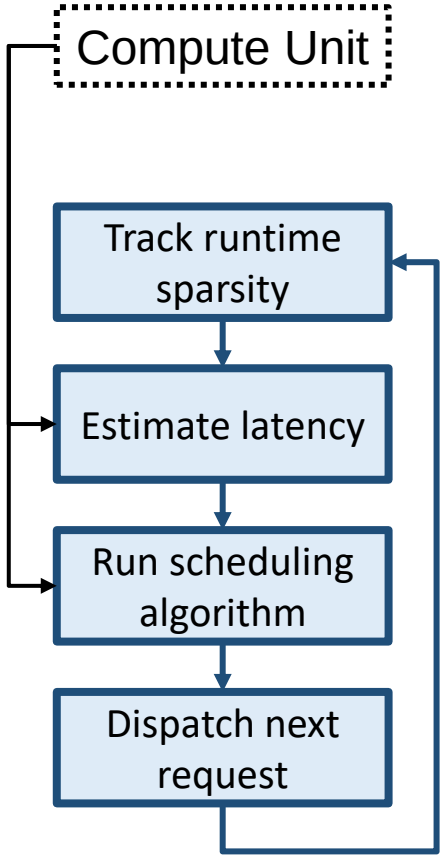


Case Study: Dysta – Enhancing existing NPUs

SAMSUNG AI Center - Cambridge

- Hardware scheduler
  - FIFOs: Track per-task information
  - Monitor: Track runtime sparsity
  - Compute unit: Estimate latency and update schedule
  - Reconfigurable design: Reuse resources for latency estimation and scheduling algorithm
- Up to 10% lower latency constraint violation rate and nearly 4x lower average normalised turnaround time across request traffic levels over state-of-the-art methods

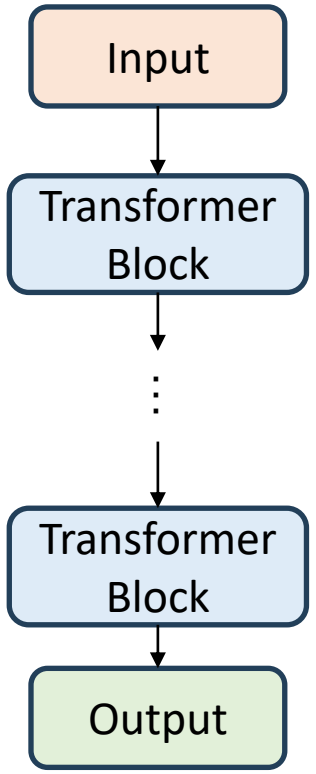




Future Trends & Open Questions

SAMSUNG AI Center - Cambridge

- Foundation models
  - Powerful pretrained models
  - Parameter-efficient Fine-tuning (PEFT) to downstream tasks
  - **Frozen shared** part & **task-specific adaptation** part

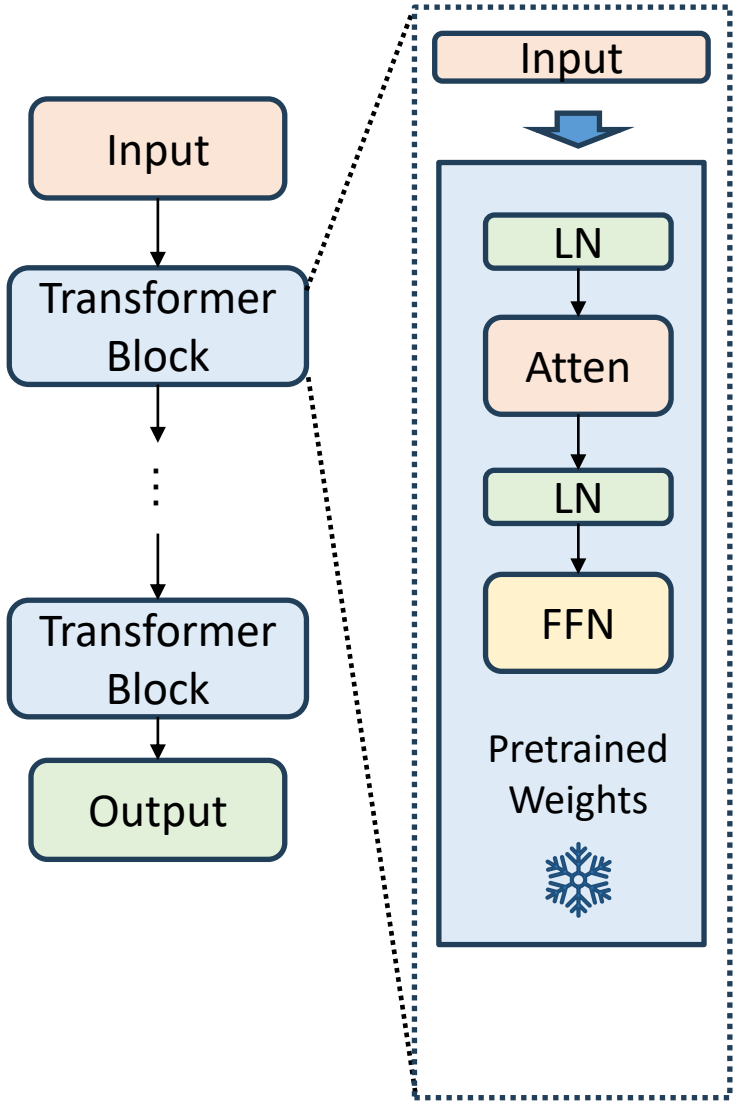


[2] L. Chen, et al. Punica: Multi-Tenant LoRA Serving. *MLSys,* 2024.
[3] Sheng, Ying, et al.S-LoRA: Serving Thousands of Concurrent LoRA Adapters. *MLSys, 2024.*

Future Trends & Open Questions

SAMSUNG AI Center - Cambridge

- Foundation models
  - Powerful pretrained models
  - Parameter-efficient Fine-tuning (PEFT) to downstream tasks
  - **Frozen shared** part & **task-specific adaptation** part

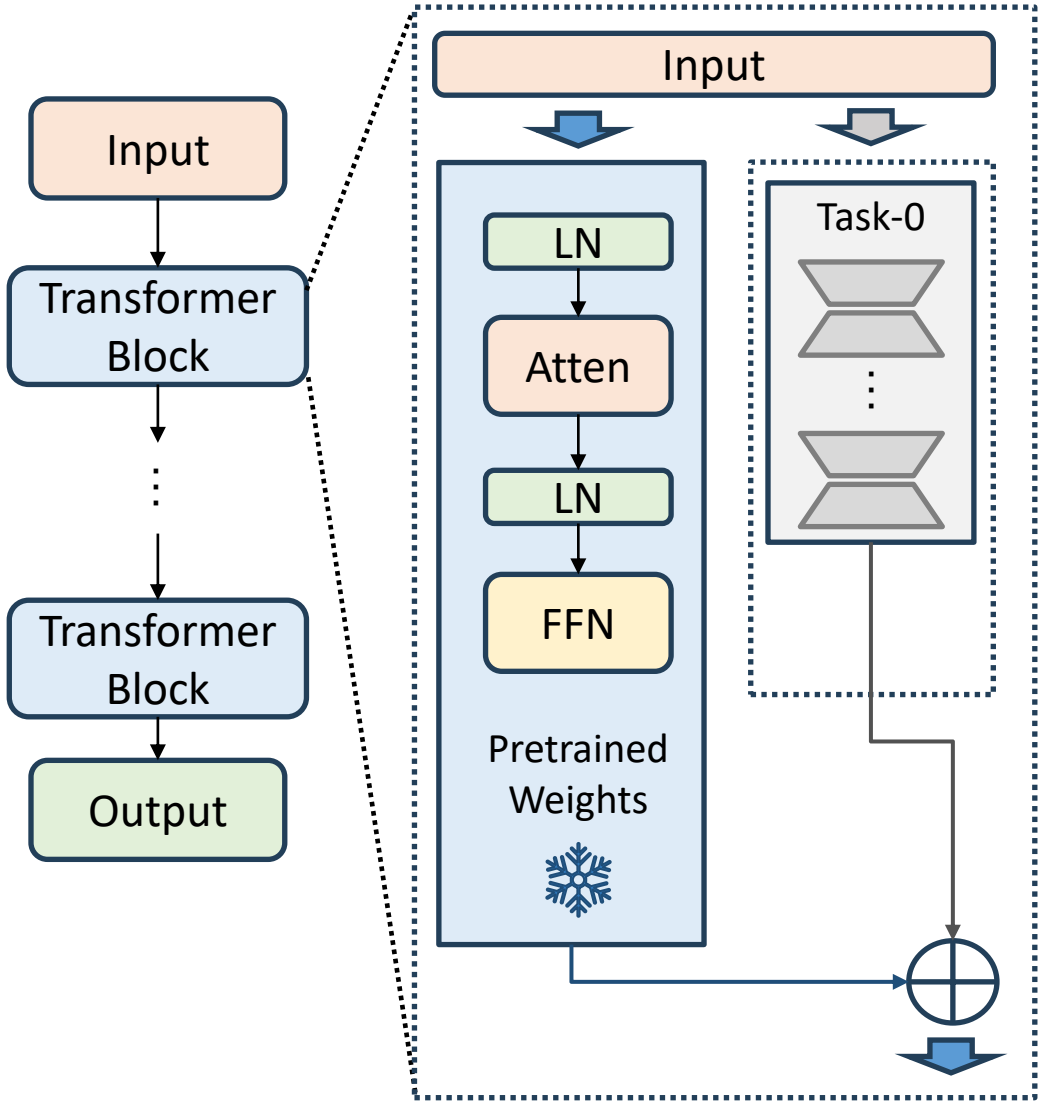


[2] L. Chen, et al. Punica: Multi-Tenant LoRA Serving. *MLSys,* 2024.
[3] Sheng, Ying, et al.S-LoRA: Serving Thousands of Concurrent LoRA Adapters. *MLSys, 2024.*

Future Trends & Open Questions

SAMSUNG AI Center - Cambridge

- Foundation models
  - Powerful pretrained models
  - Parameter-efficient Fine-tuning (PEFT) to downstream tasks
  - **Frozen shared** part & **task-specific adaptation** part

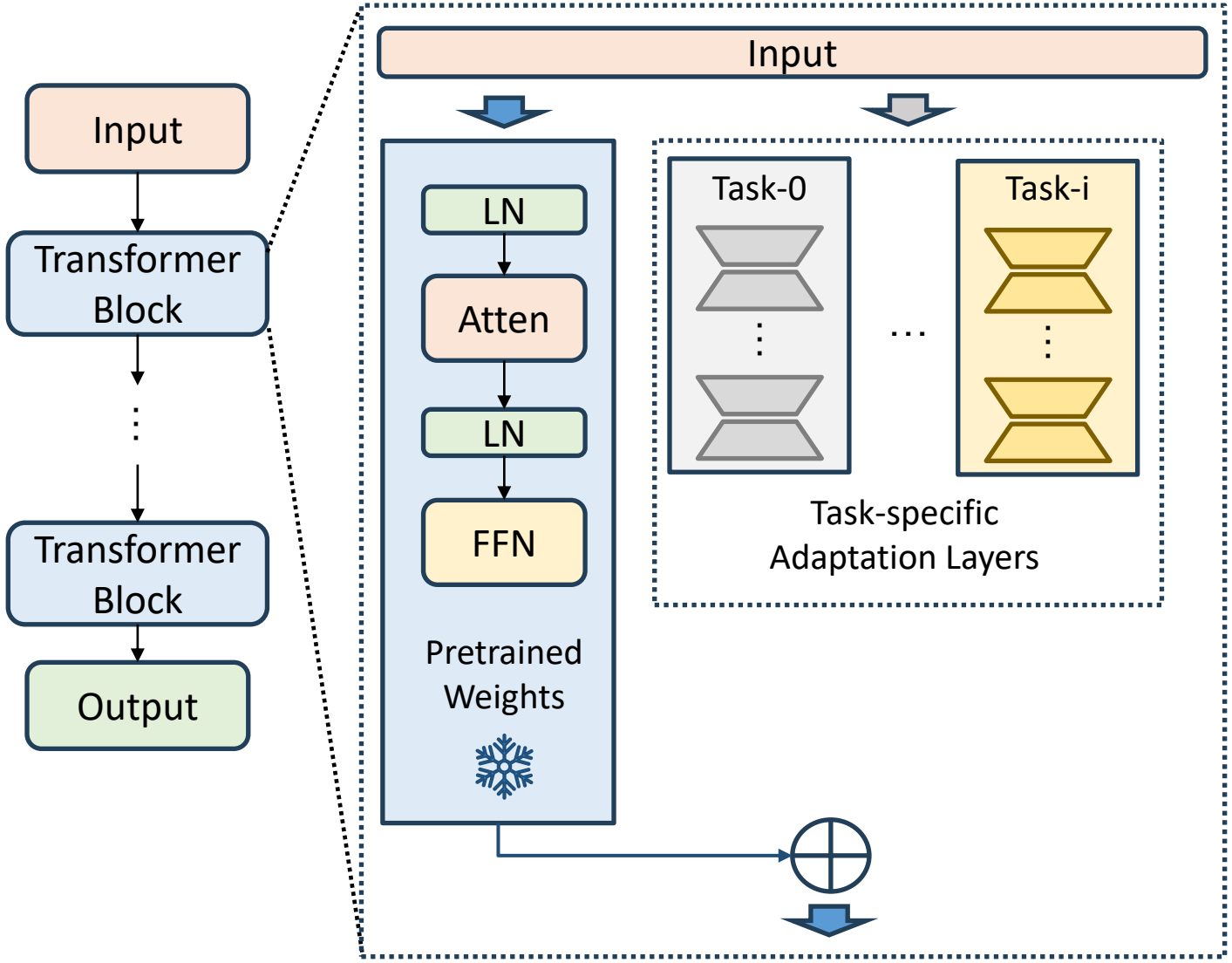


[2] L. Chen, et al. Punica: Multi-Tenant LoRA Serving. *MLSys,* 2024.
[3] Sheng, Ying, et al.S-LoRA: Serving Thousands of Concurrent LoRA Adapters. *MLSys, 2024.*

Future Trends & Open Questions

SAMSUNG AI Center - Cambridge

- Foundation models
  - Powerful pretrained models
  - Parameter-efficient Fine-tuning (PEFT) to downstream tasks
  - **Frozen shared** part & **task-specific adaptation** part

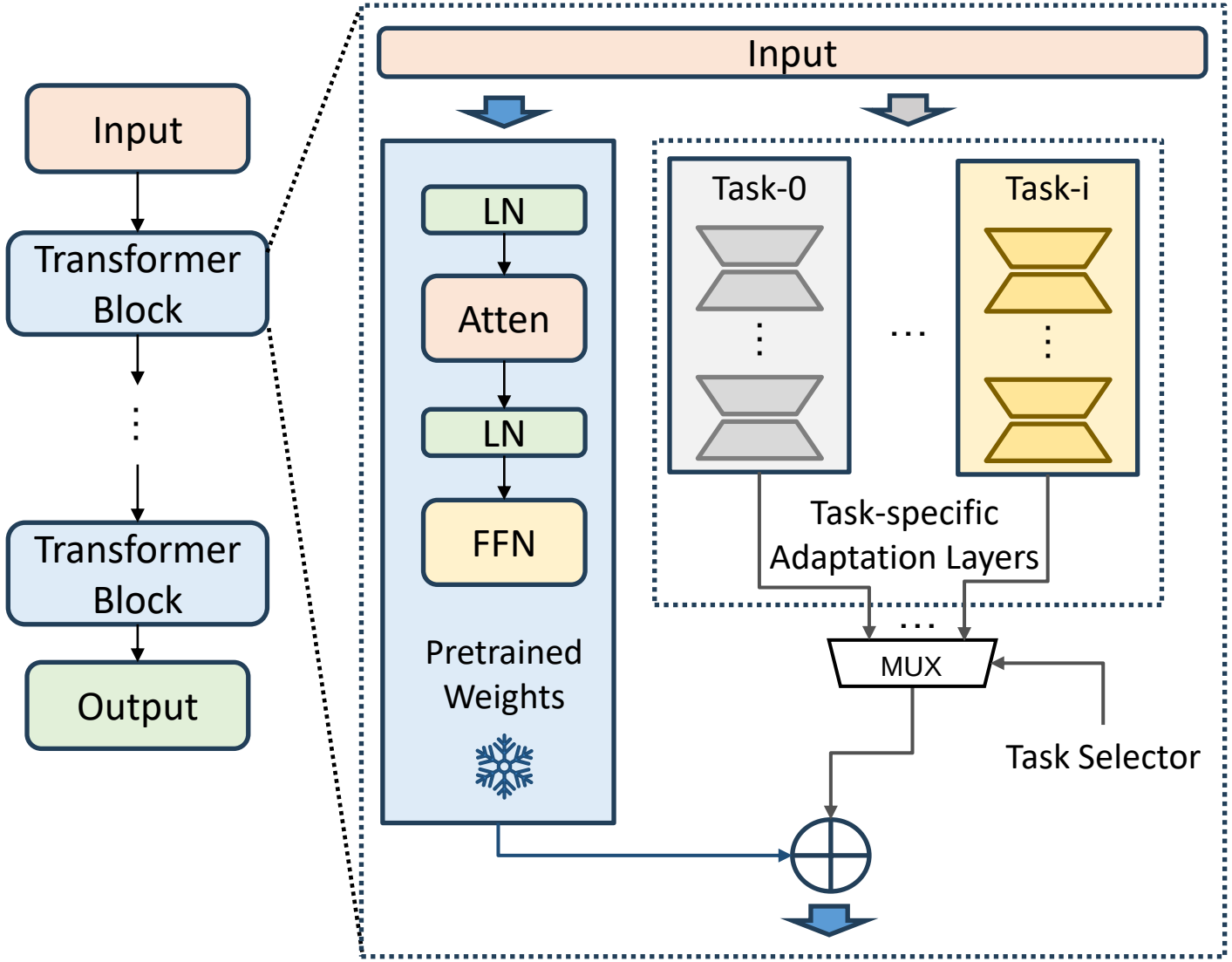


[2] L. Chen, et al. Punica: Multi-Tenant LoRA Serving. *MLSys,* 2024.
[3] Sheng, Ying, et al.S-LoRA: Serving Thousands of Concurrent LoRA Adapters. *MLSys, 2024.*

Future Trends & Open Questions

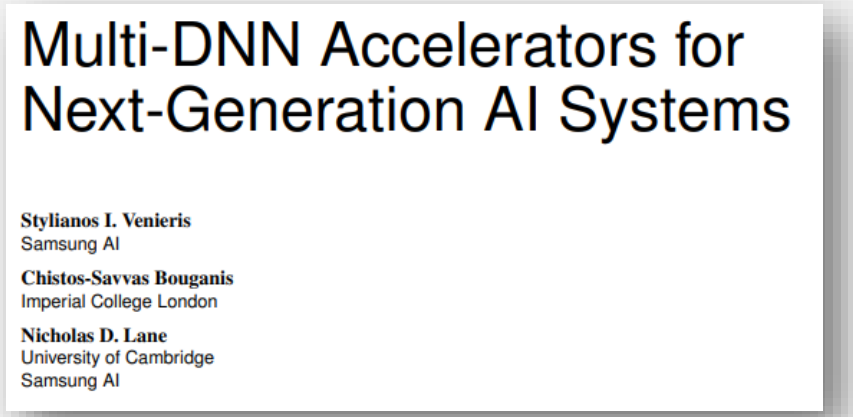
SAMSUNG AI Center - Cambridge

- Foundation models
  - Powerful pretrained models
  - Parameter-efficient Fine-tuning (PEFT) to downstream tasks
  - **Frozen shared** part & **task-specific adaptation** part
- Research Questions
  - How to swap efficiently between Adaptation layers? Different memory management approach?
  - How to design the underlying hardware? Specialised units or unified for Shared and Adaptation parts?
  - Model-Hardware Co-Design for the next milestone



[2] L. Chen, et al. Punica: Multi-Tenant LoRA Serving. *MLSys,* 2024.
[3] Sheng, Ying, et al.S-LoRA: Serving Thousands of Concurrent LoRA Adapters. *MLSys, 2024.*

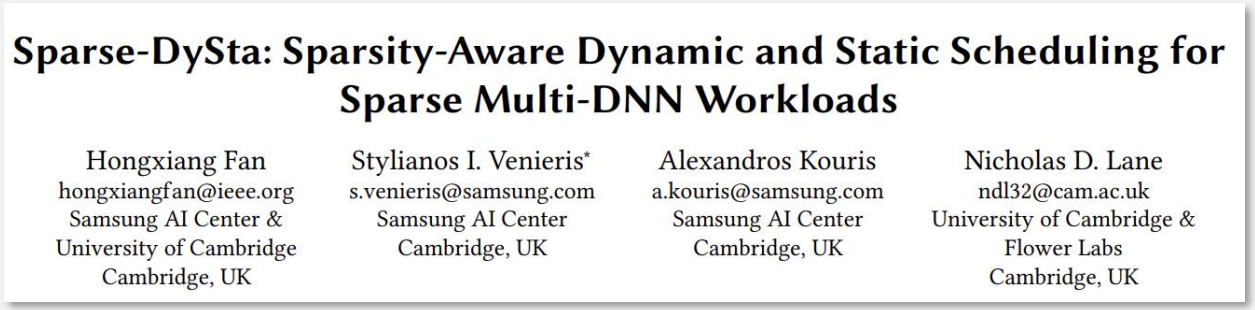
Thank you


Multi-DNN Accelerators for Next-Generation AI Systems

Stylianos I. Venieris
Samsung AI

Chistos-Savvas Bouganis
Imperial College London

Nicholas D. Lane
University of Cambridge
Samsung AI


Sparse-DySta: Sparsity-Aware Dynamic and Static Scheduling for Sparse Multi-DNN Workloads

Hongxiang Fan
hongxiangfan@ieee.org
Samsung AI Center &
University of Cambridge
Cambridge, UK

Stylianos I. Venieris*
s.venieris@samsung.com
Samsung AI Center
Cambridge, UK

Alexandros Kouris
a.kouris@samsung.com
Samsung AI Center
Cambridge, UK

Nicholas D. Lane
ndl32@cam.ac.uk
University of Cambridge &
Flower Labs
Cambridge, UK

Stylianos I. Venieris

Samsung AI

s.venieris@samsung.com | @SIVenieris