

# Summarizer:

## Trading Communication with Computing Near Storage

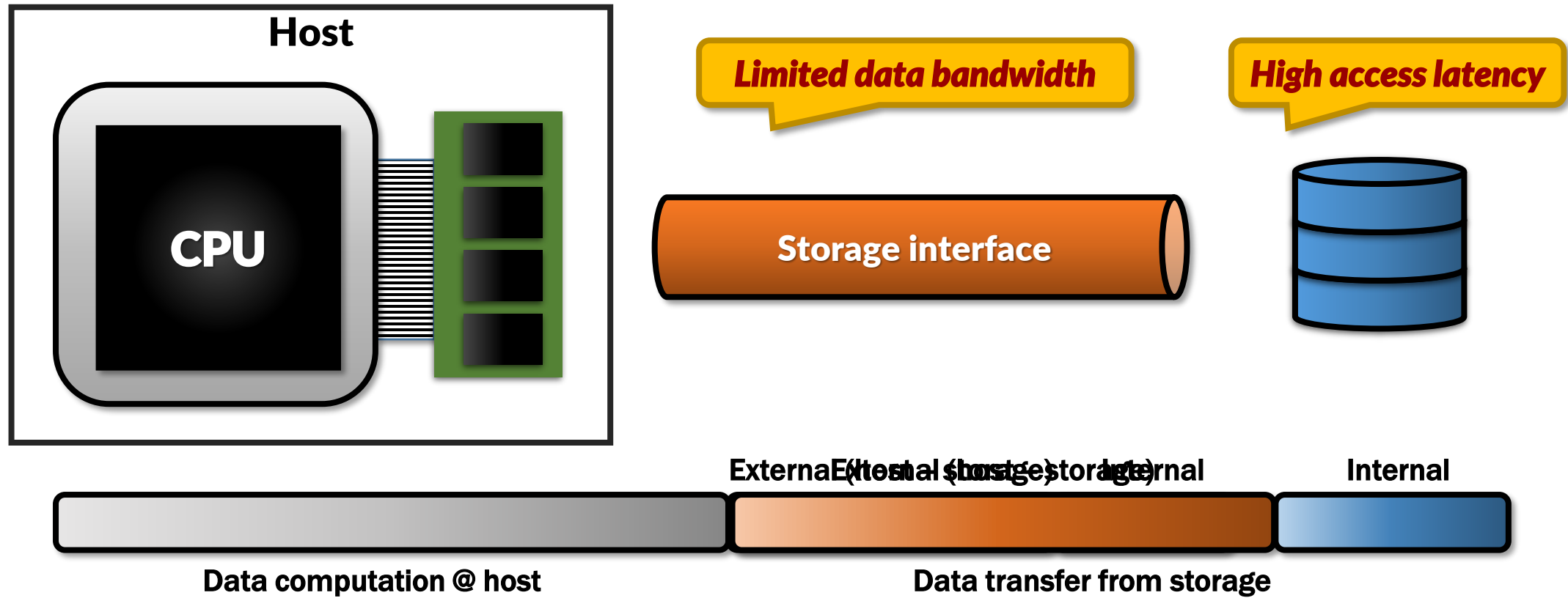
Gunjae Koo<sup>\*</sup>, Kiran Kumar Matam<sup>\*</sup>, Te I<sup>†</sup>, H.V. Krishina Giri Nara<sup>\*</sup>, Jing Li<sup>‡</sup>,  
Hung-Wei Tseng<sup>†</sup>, Steven Swanson<sup>‡</sup>, Murali Annavaram<sup>\*</sup>

<sup>\*</sup>University of Southern California

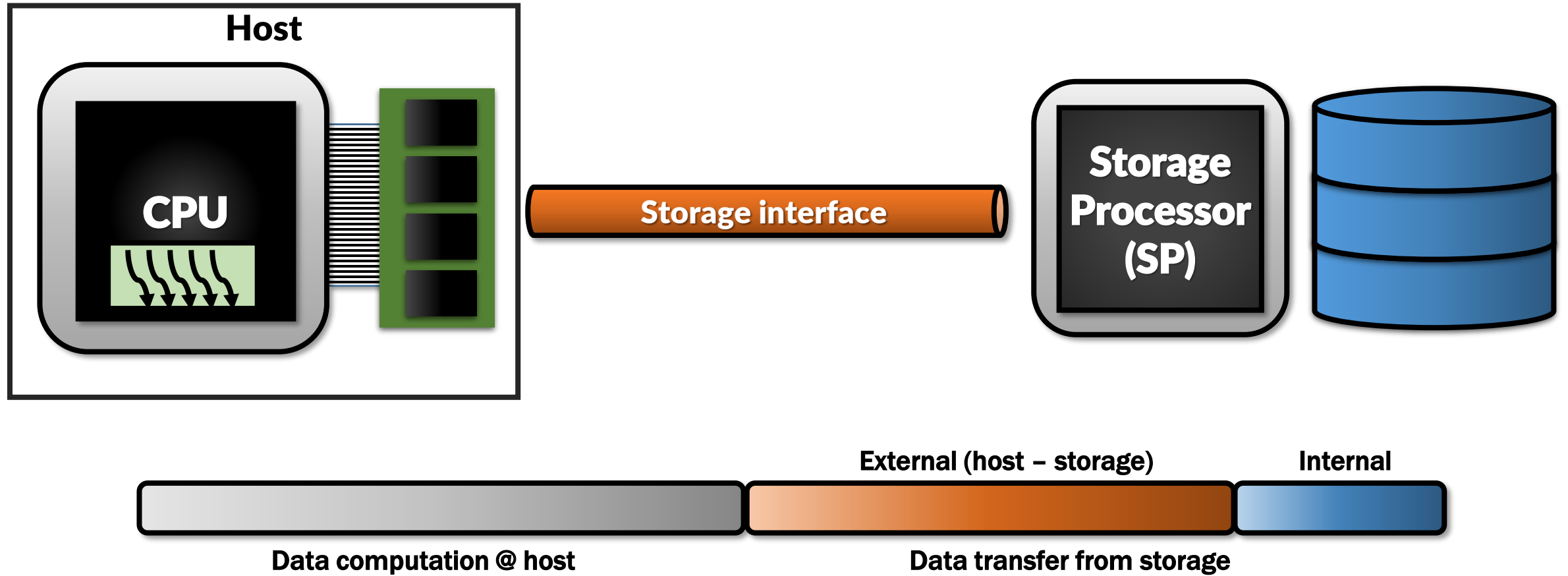
<sup>†</sup>North Carolina State University

<sup>‡</sup>University of California, San Diego

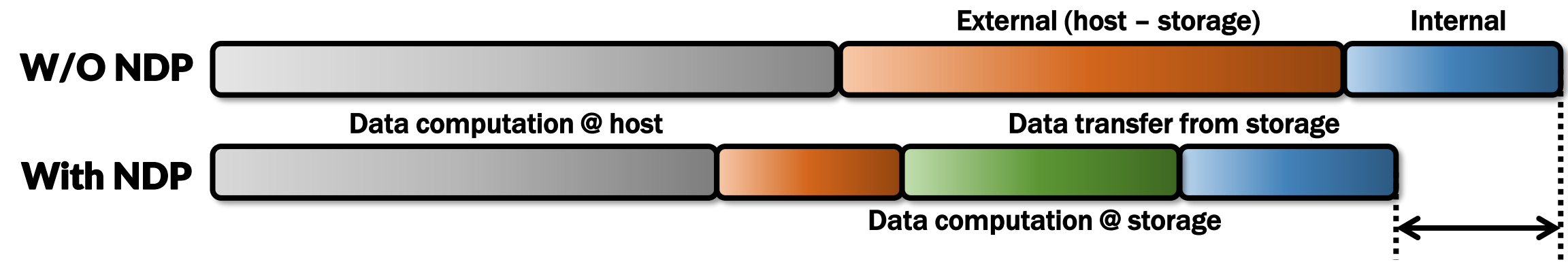
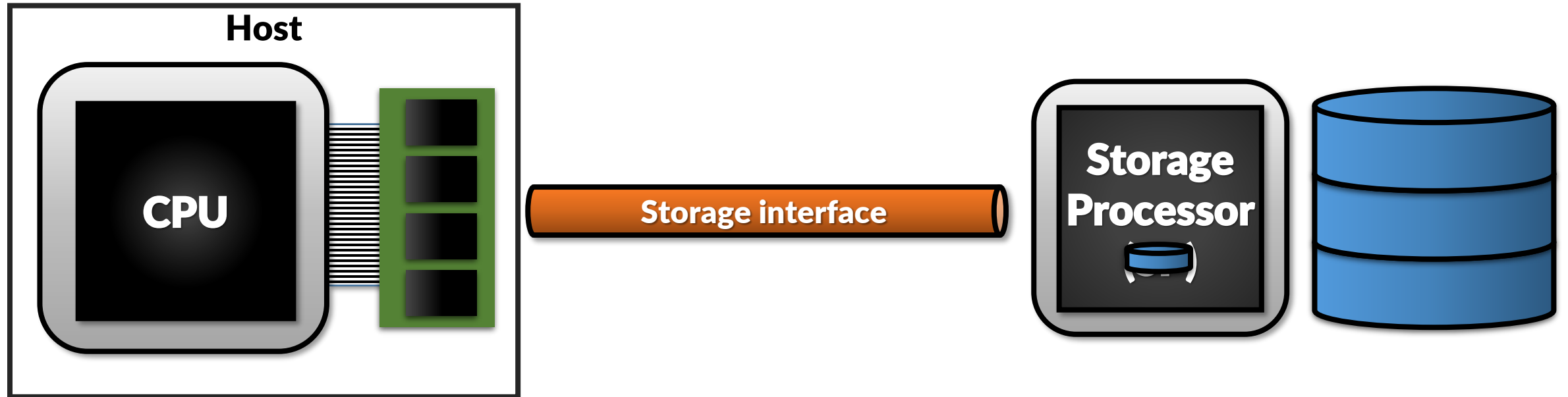
# Motivation – High Data Movement Cost



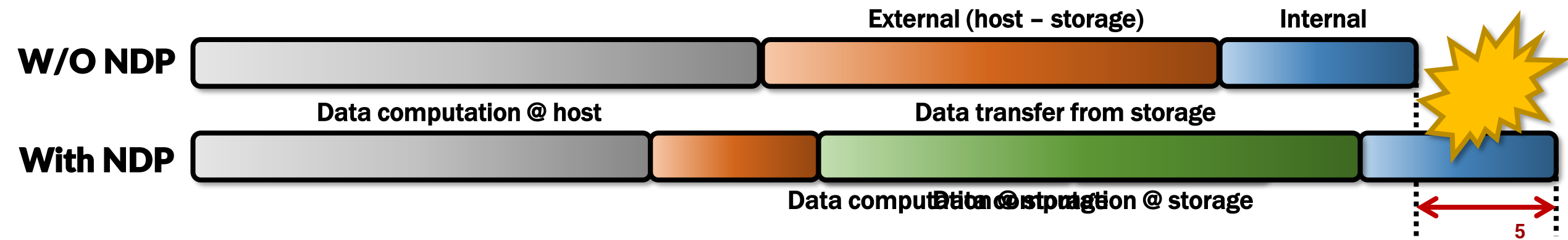
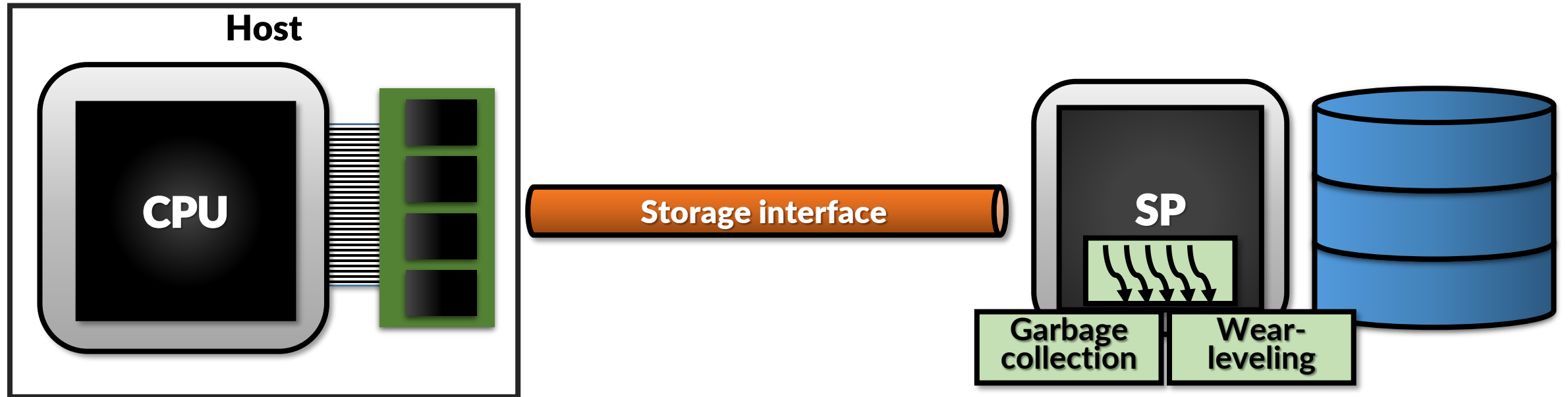
# Near Data Processing (NDP)



# Near Data Processing (NDP)



# Near Data Processing (NDP) on SSDs



Host

## **Obstacles to in-SSD processing**

- **Less powerful embedded processor**
- **Dynamic computation resource availability**

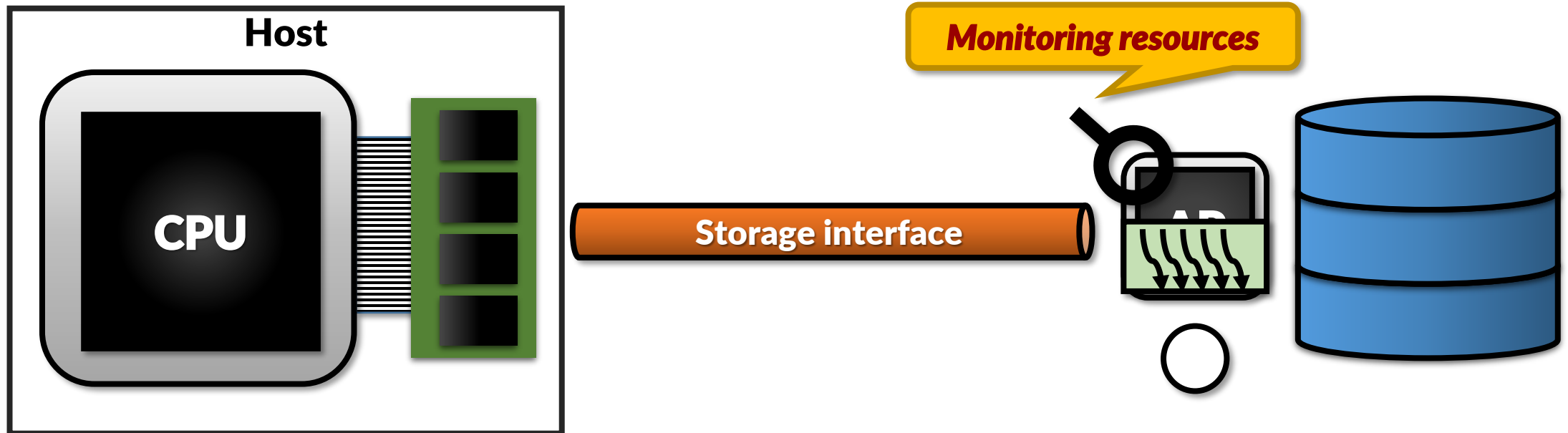
**Summarizer: Dynamic NDP framework for SSD**

W/O NDP

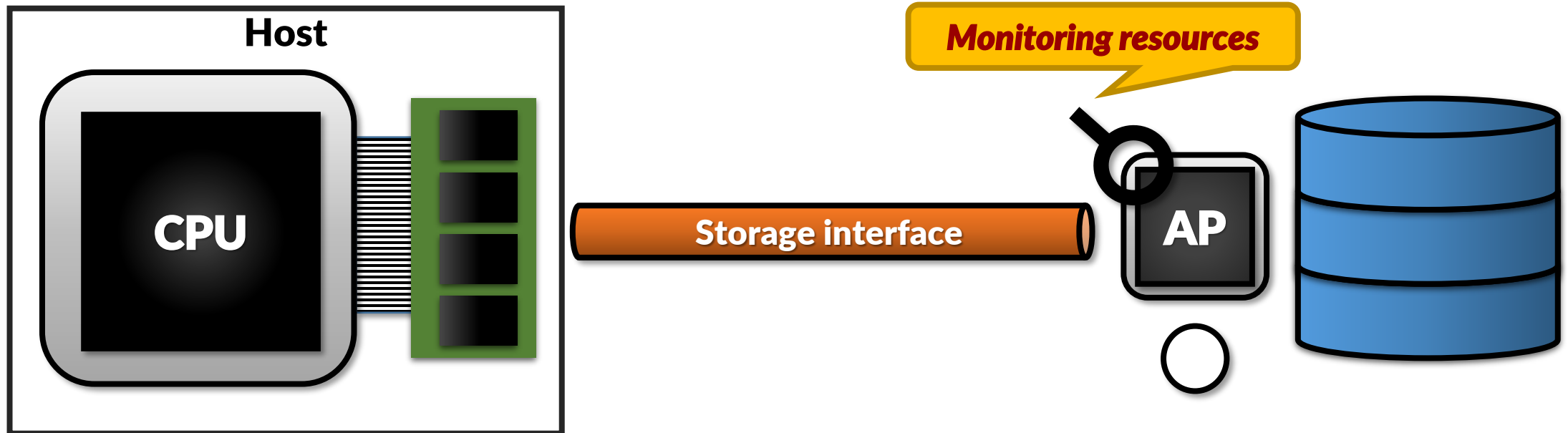
With NDP

Data computation @ storage

# Summarizer – Basic Concept

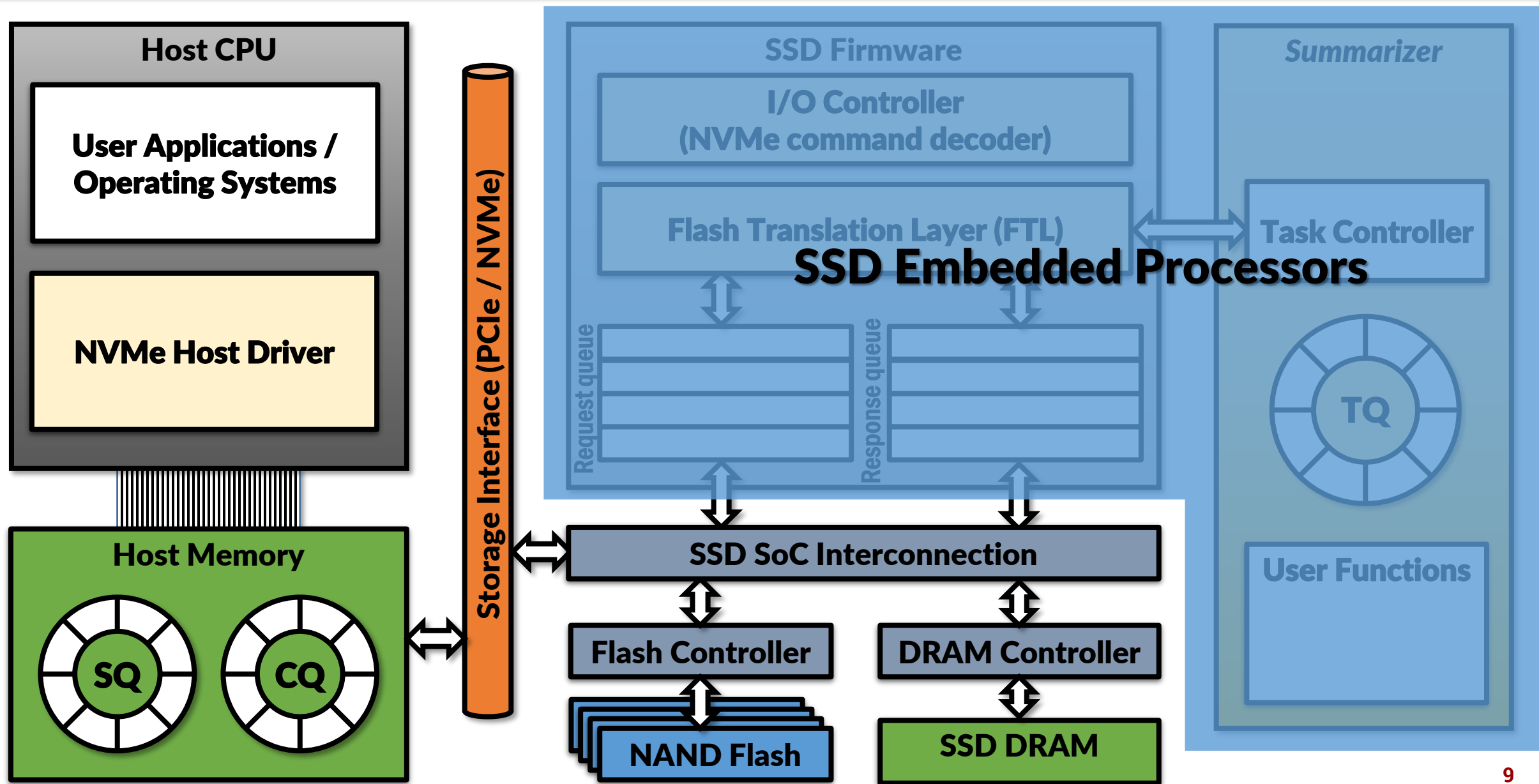


# Summarizer – Basic Concept

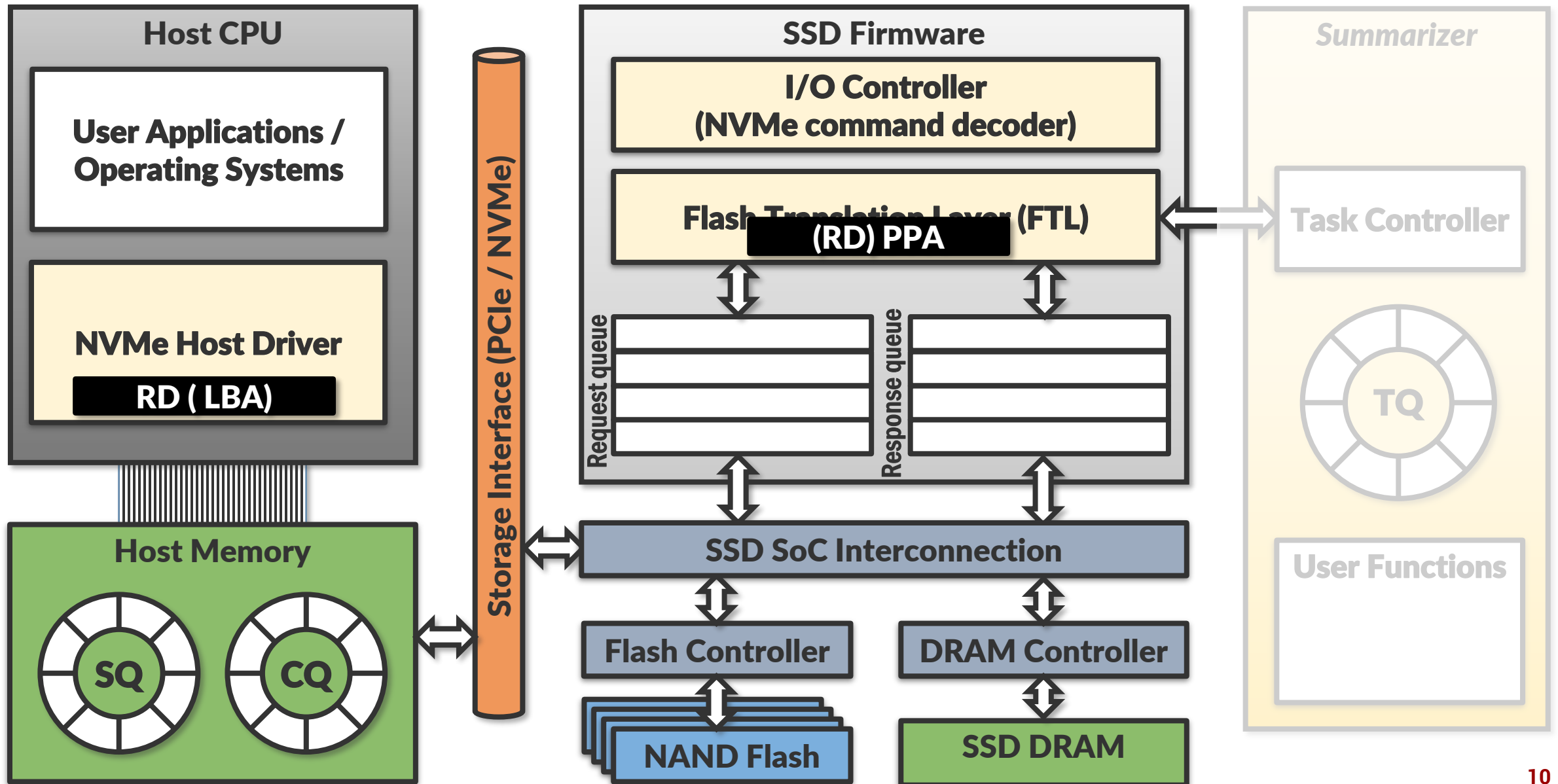




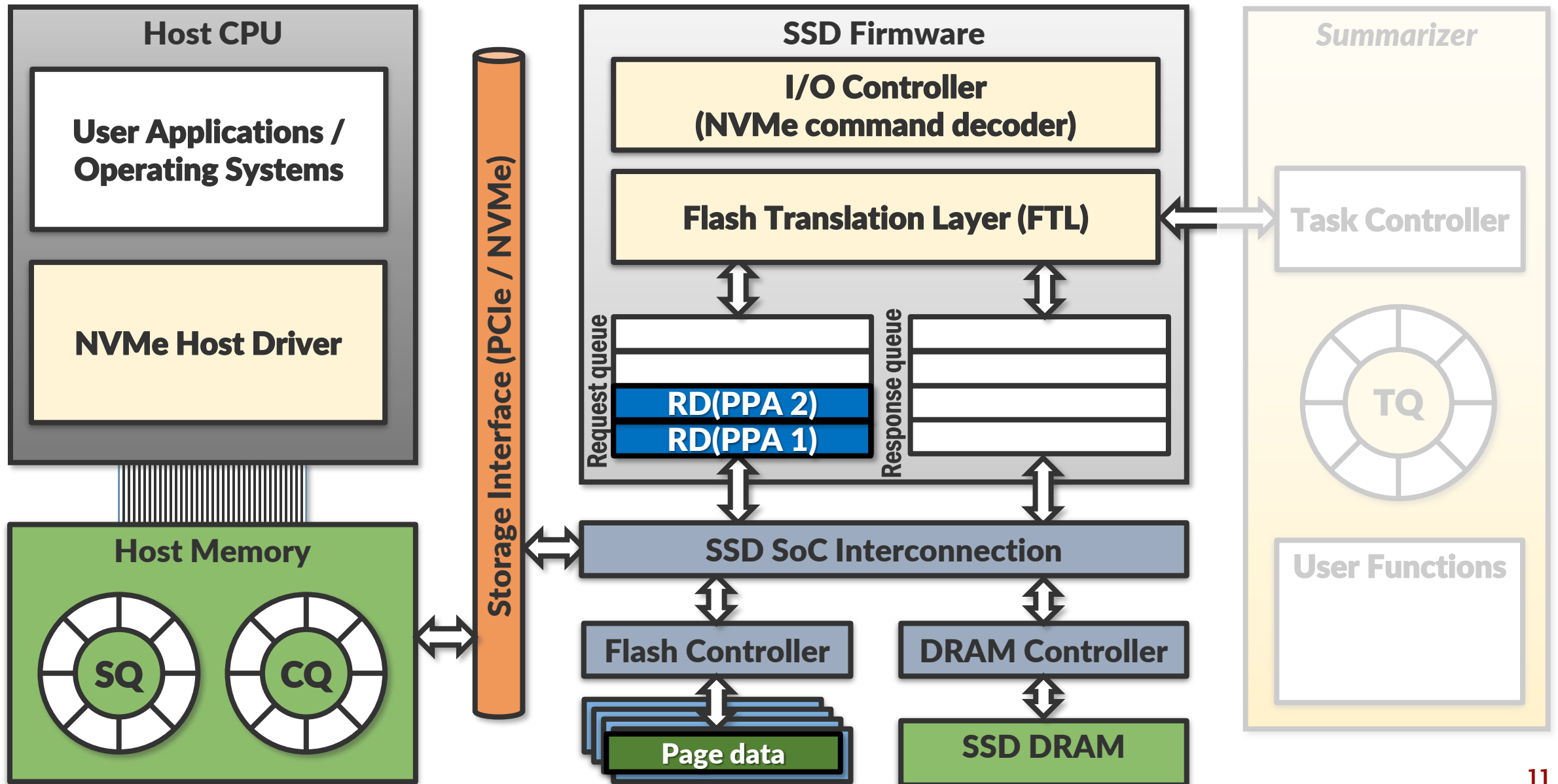
# Summarizer – Detailed Firmware Architecture



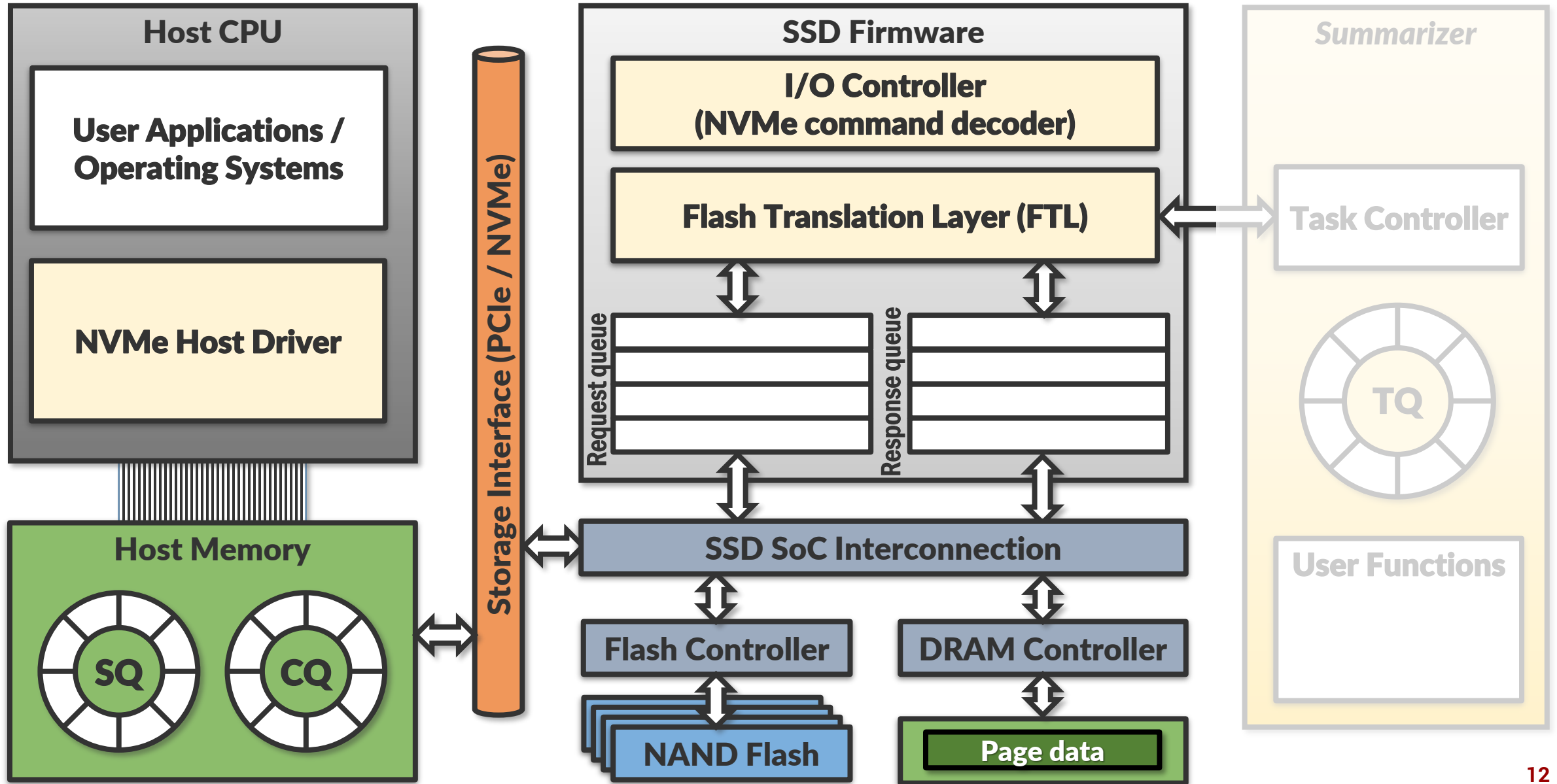
# Normal Page Read Request



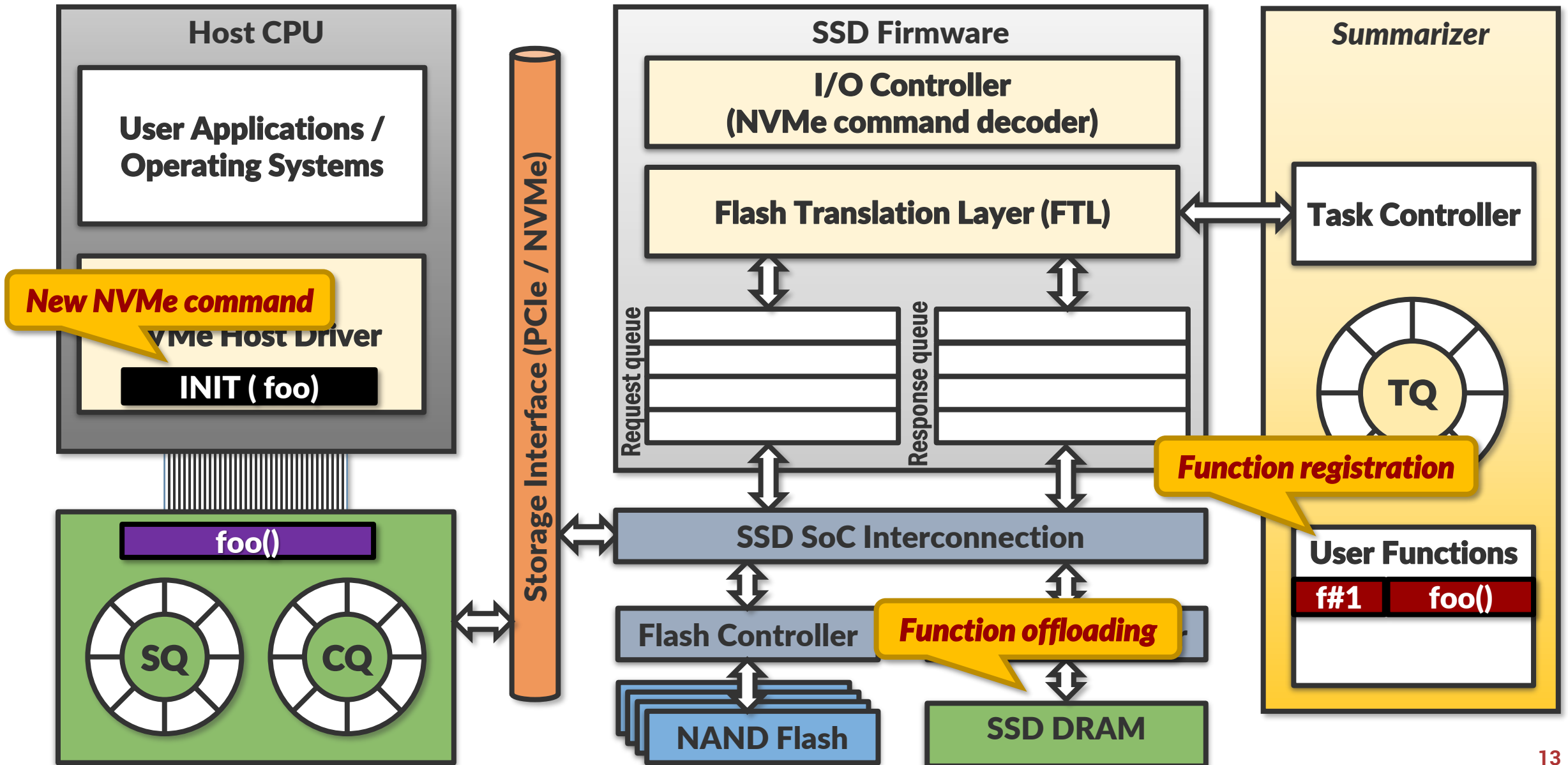
# Normal Page Read Request



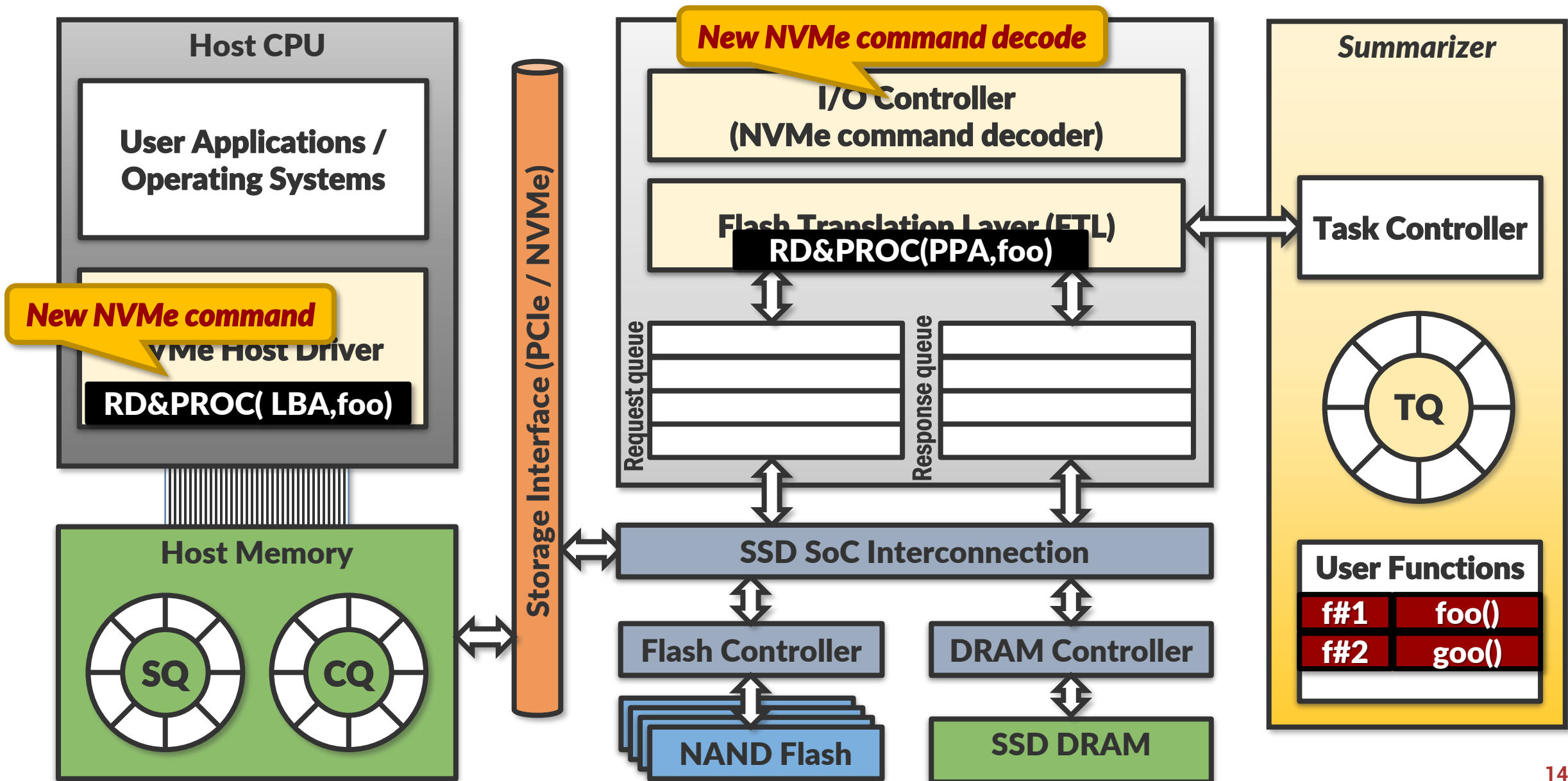
# Normal Page Read Request



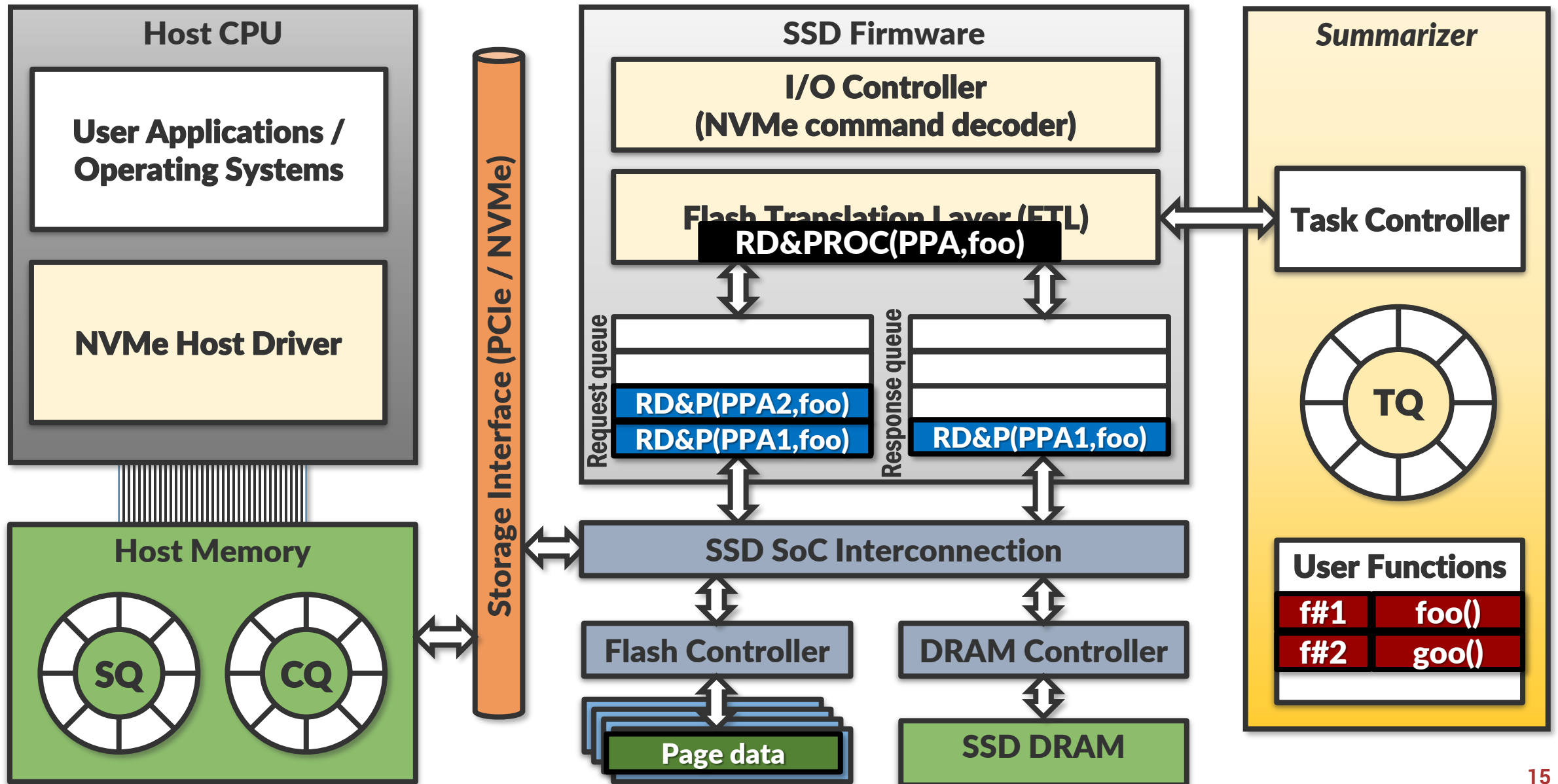
# Summarizer – Initialization (Function Offloading)



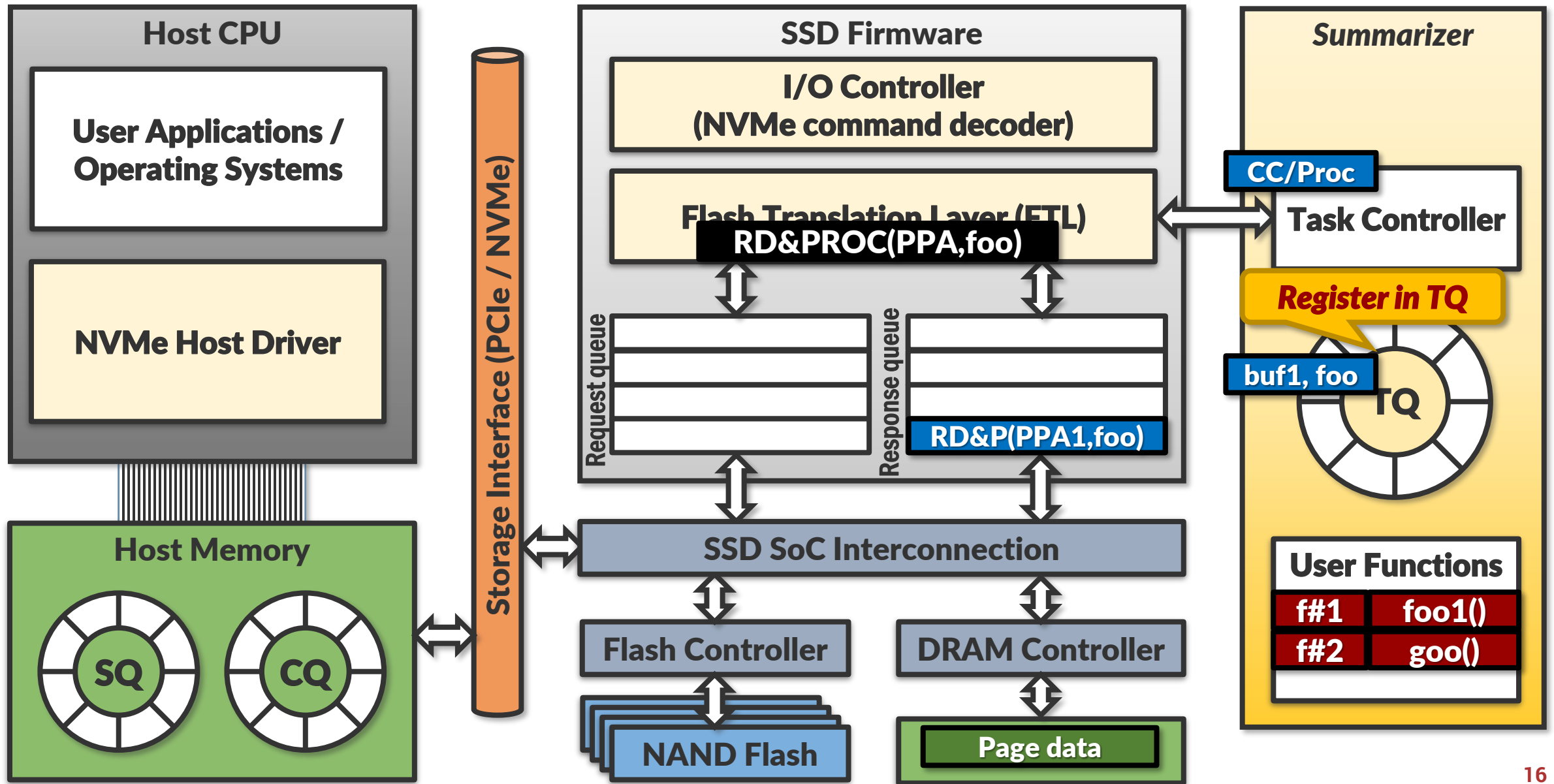
# Summarizer - Computation (Dynamic mode)



# Summarizer - Computation (Dynamic mode)

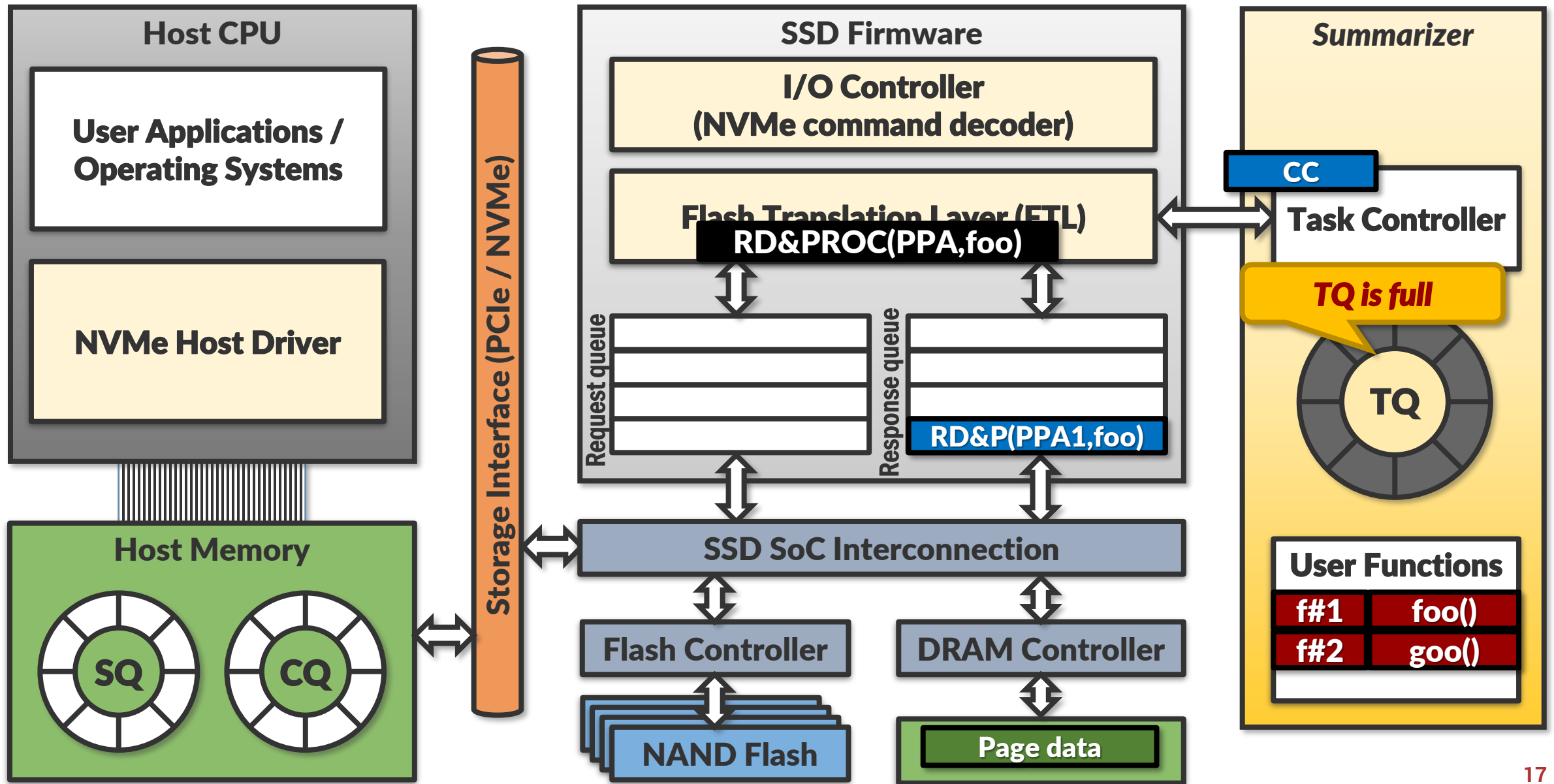


# Summarizer - Computation (Dynamic mode)

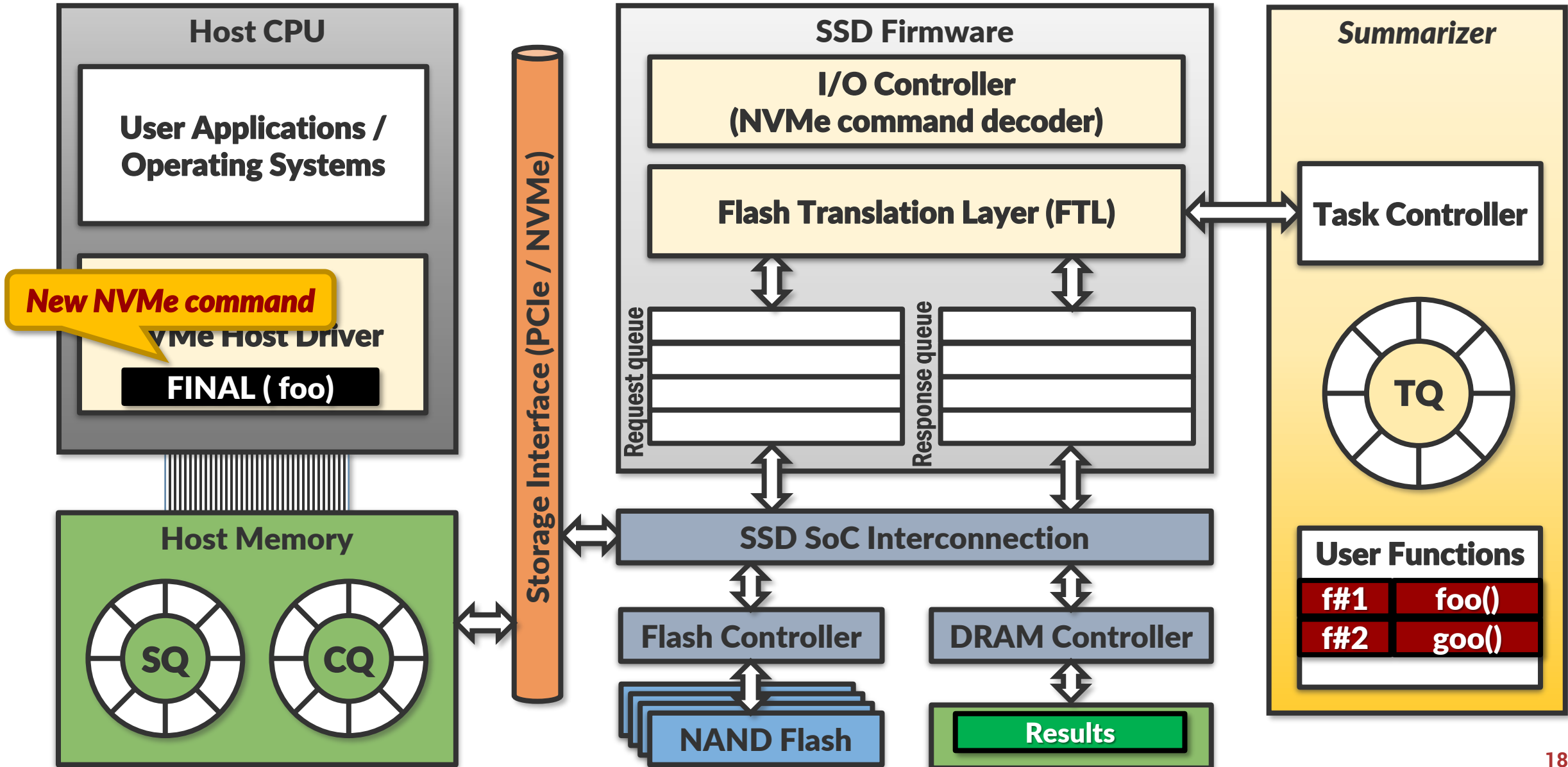




# Summarizer - Computation (Dynamic mode)



# Summarizer – Finalization



# Summarizer API and NVMe commands

## Initialization

- NVMe command: *INIT\_TSKn*
- Transfer a *in-SSD procedure* to SSD memory
- Initialize data structure and temporal variables for in-SSD computation

## Computation

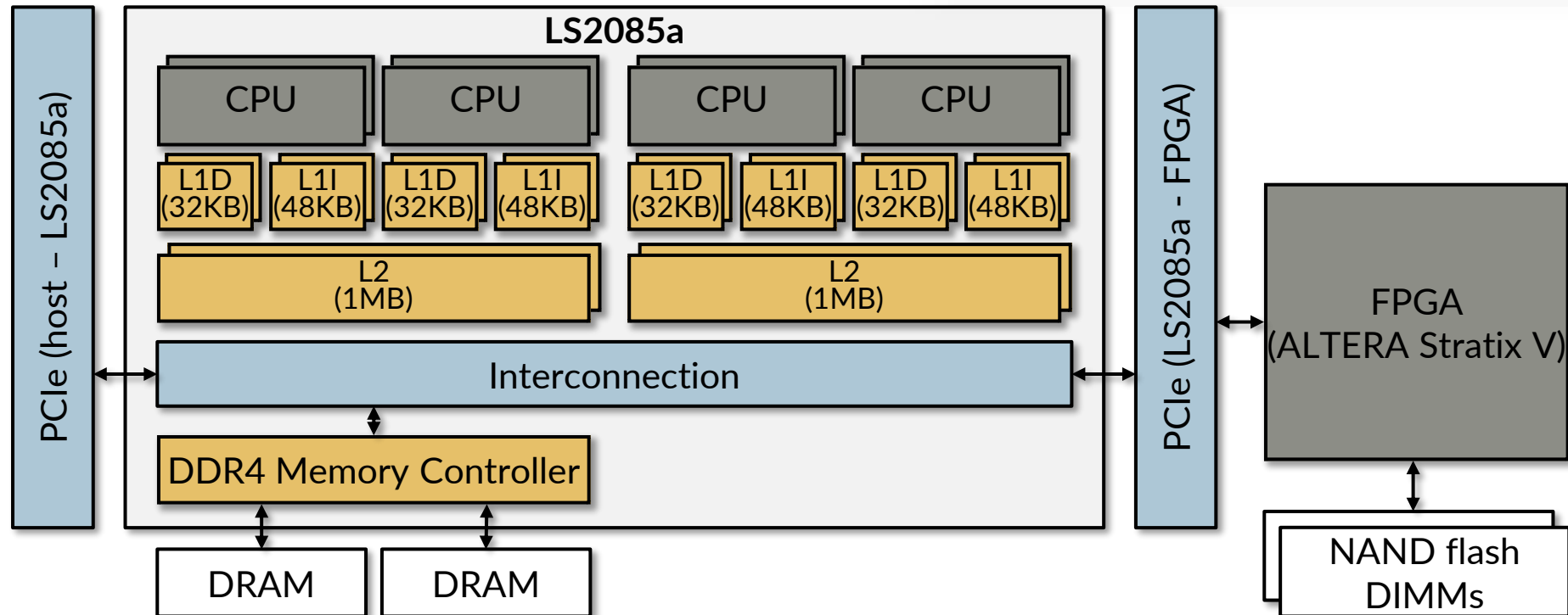
- NVMe command: *READ\_PROC\_TSKn*
- Page read command is issued with the *flag* indicating the *user procedure* embedded in SSD memory
- Return the special code if the requested page is processed in SSD
- Page data is transferred to the host if the requested page is *NOT* computed in SSD

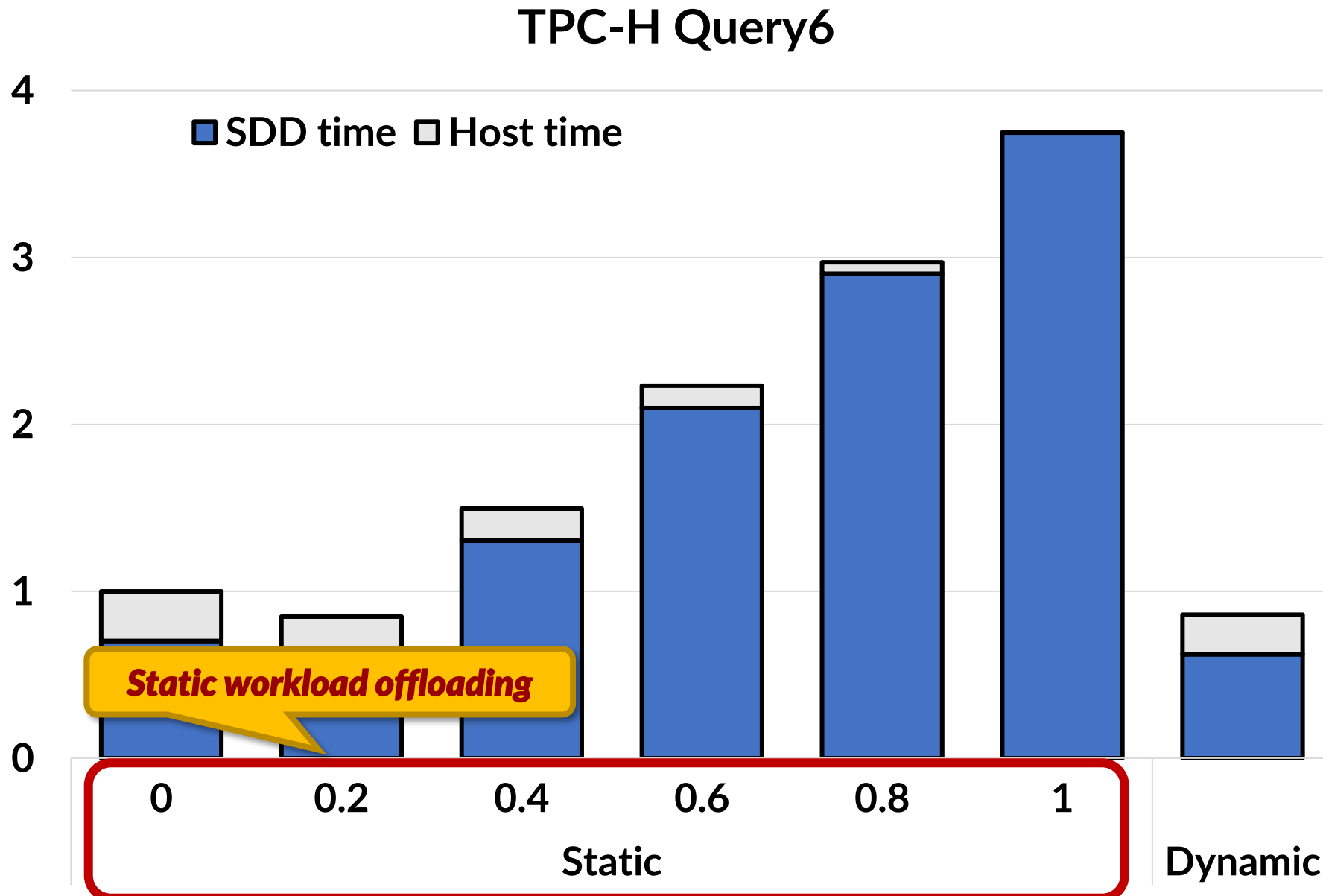
## Finalization

- NVMe command: *FINAL\_TSKn*
- Gather final in-SSD computation results and transfer to the host

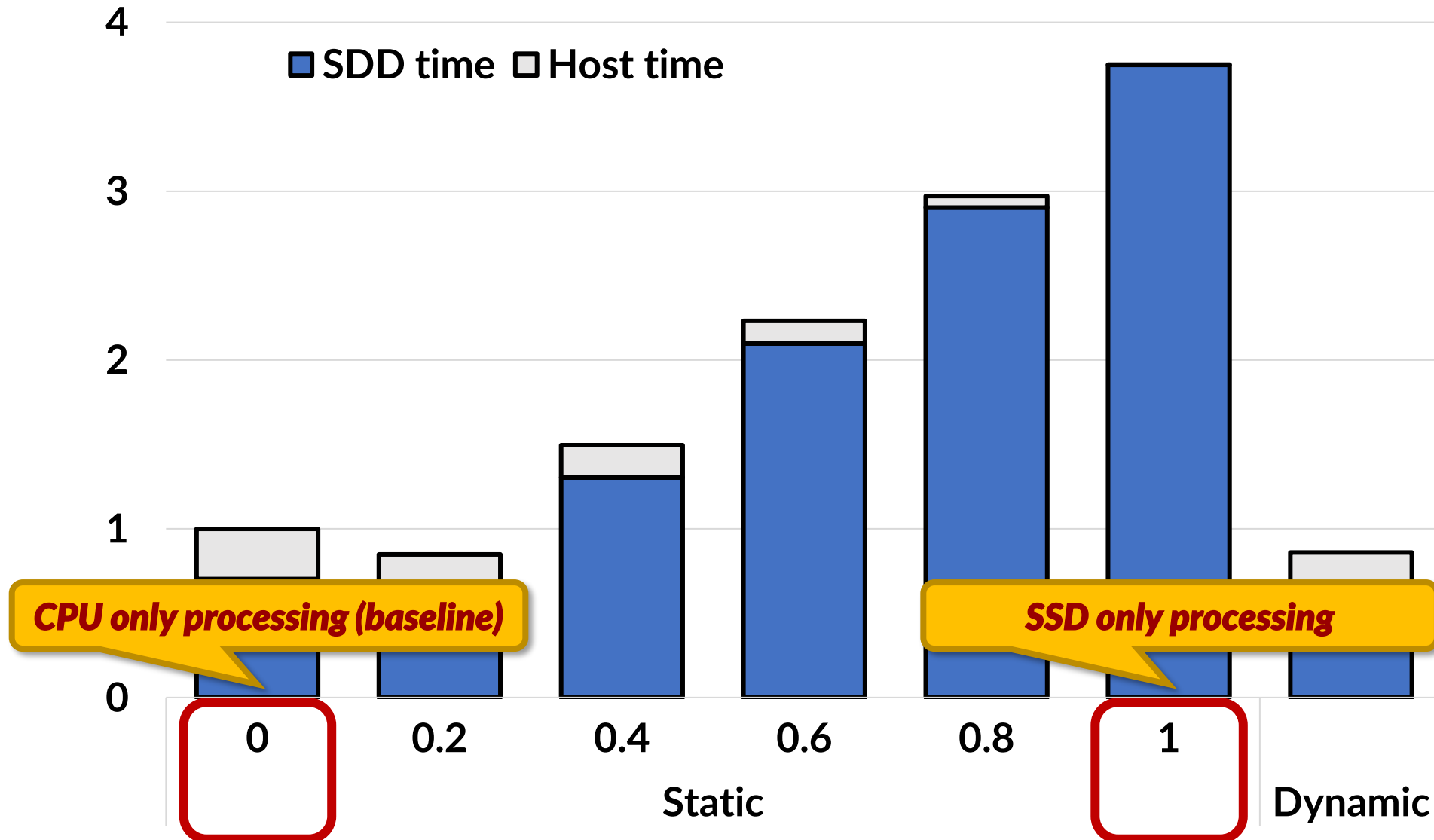
# Evaluation Platform

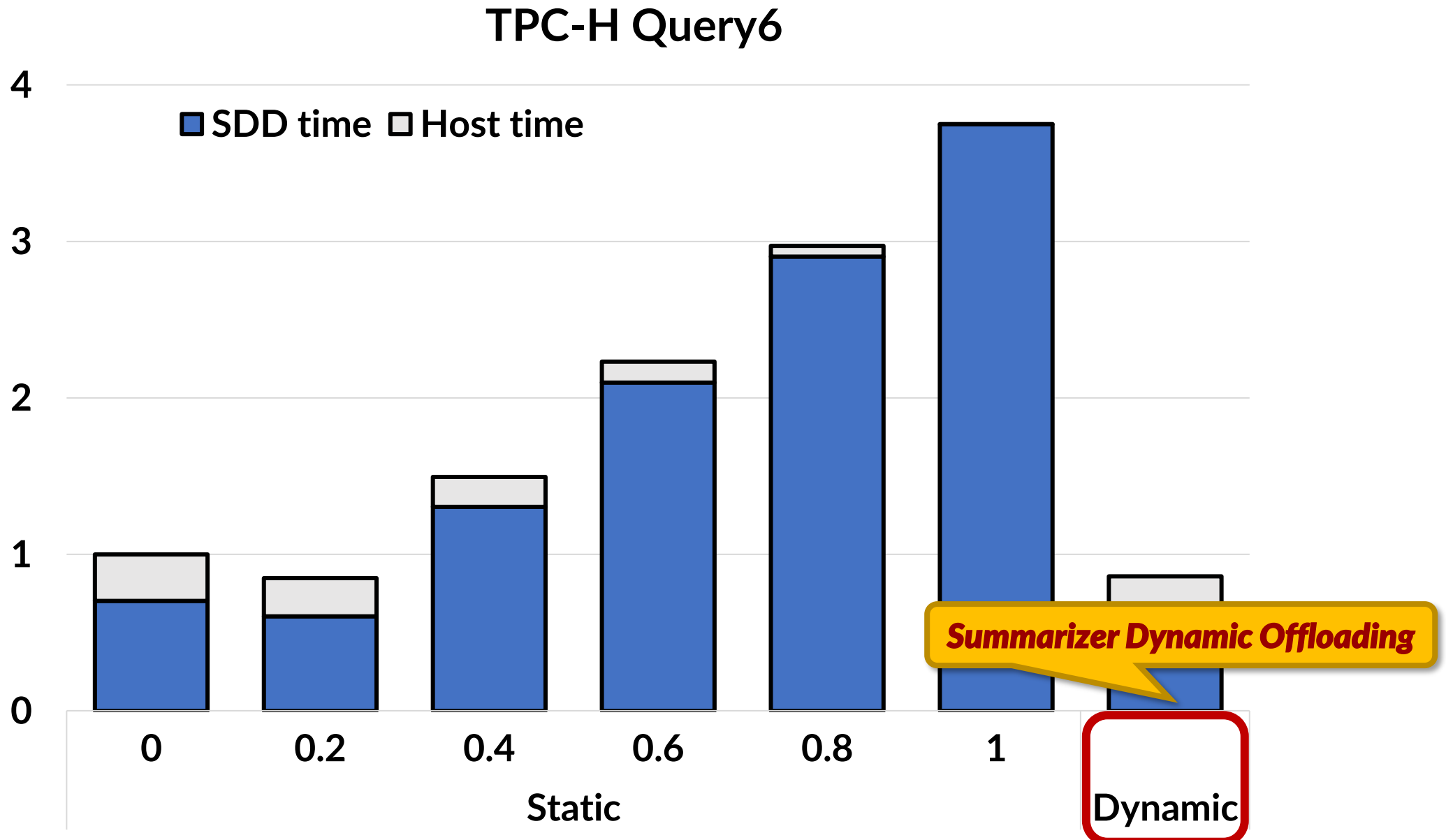
- LS2085a intelligent SSD development platform
- ARM cores running FTL and *Summarizer* firmware
- FPGA implementing NAND flash controller
- PCIe Gen. 3 4x lanes for host communication



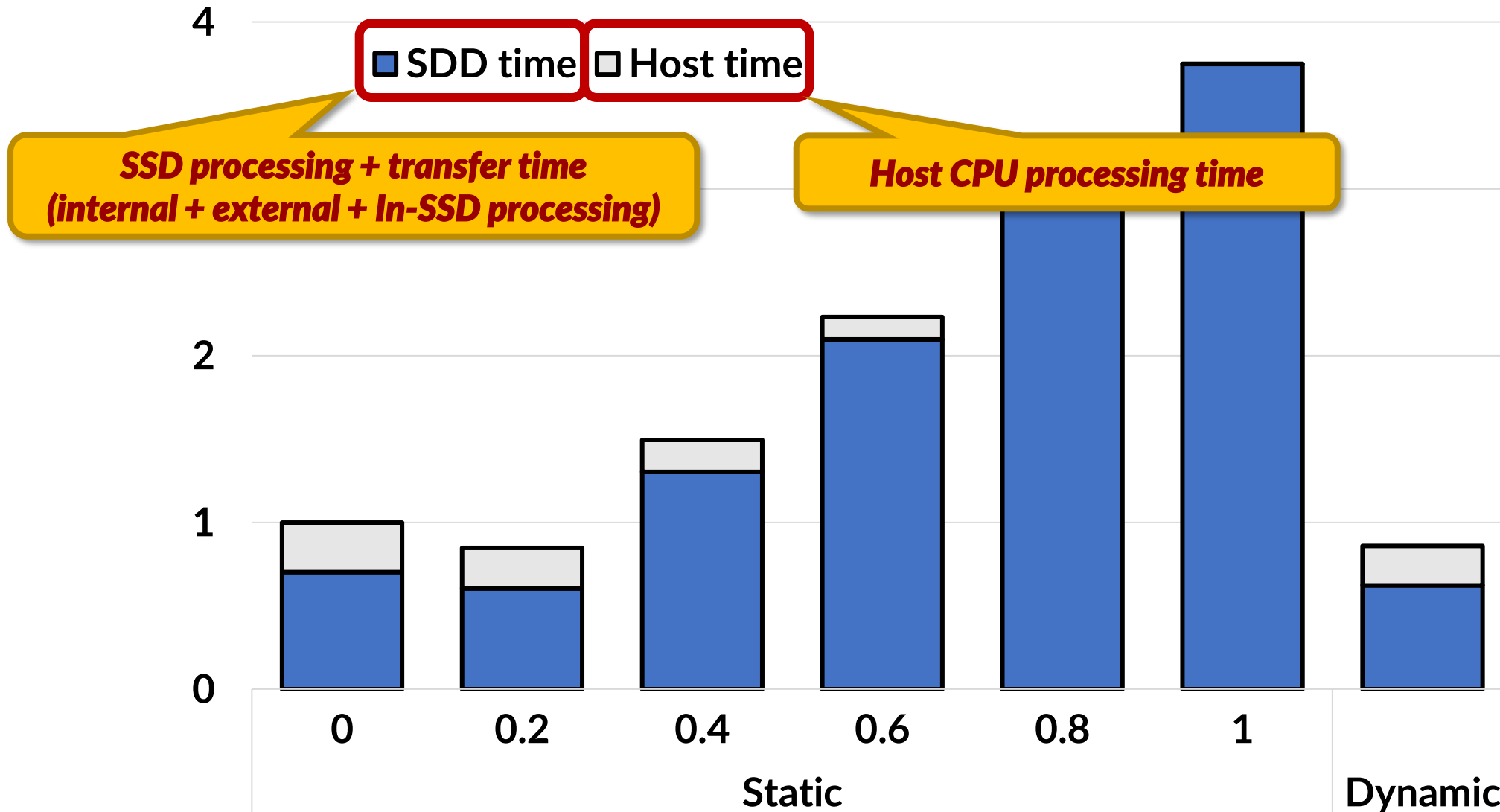


## TPC-H Query6



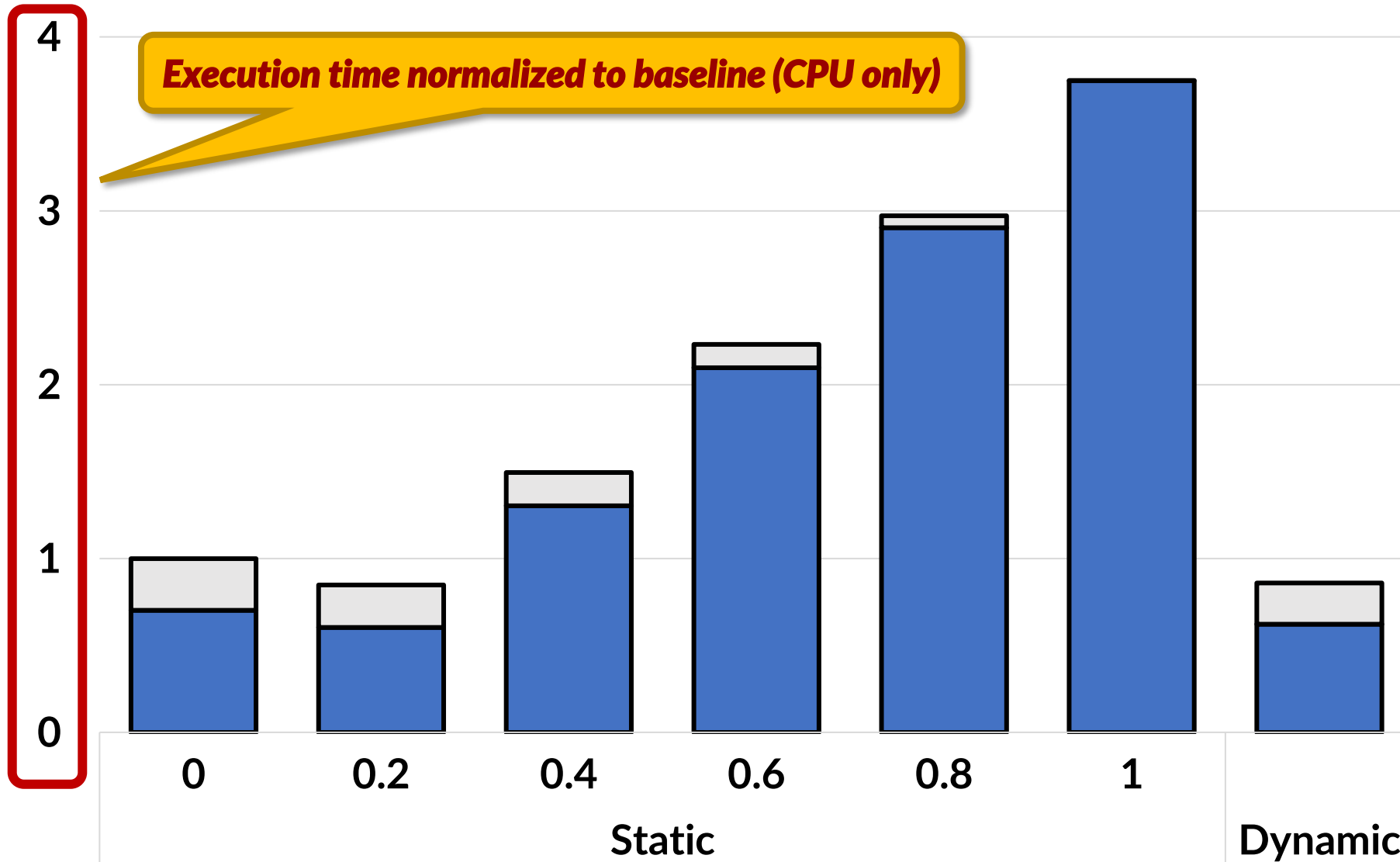


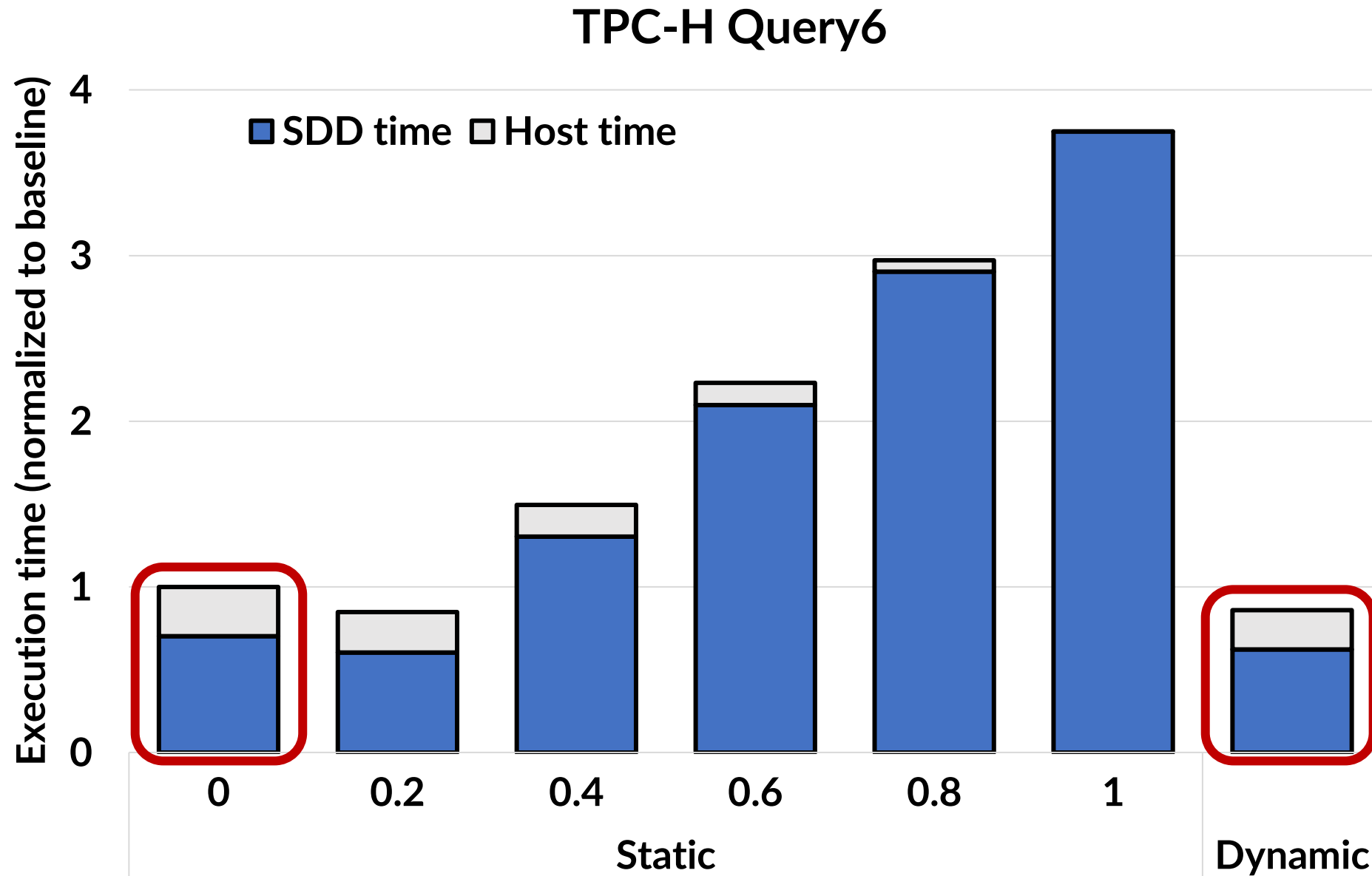
## TPC-H Query6



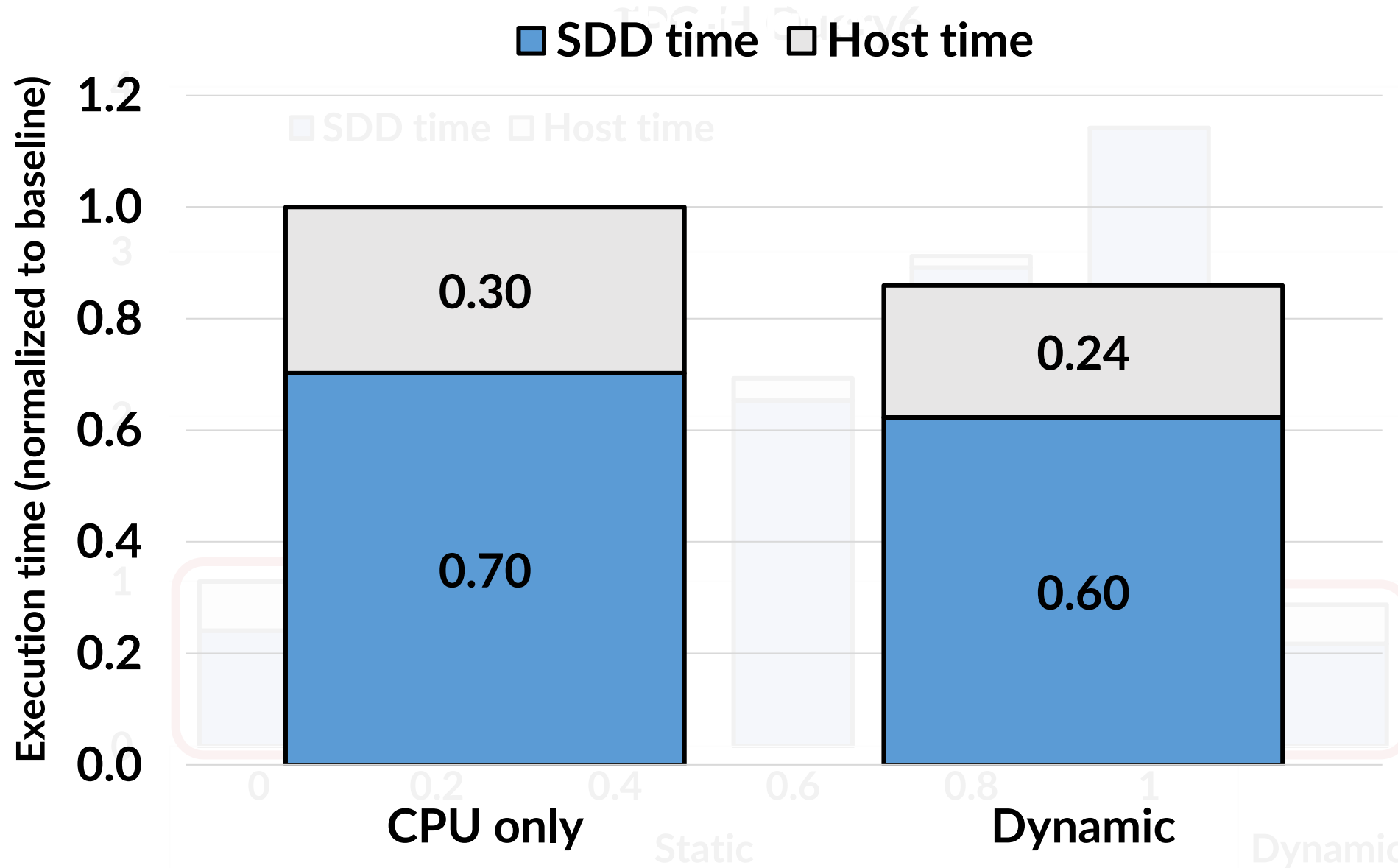


## TPC-H Query6

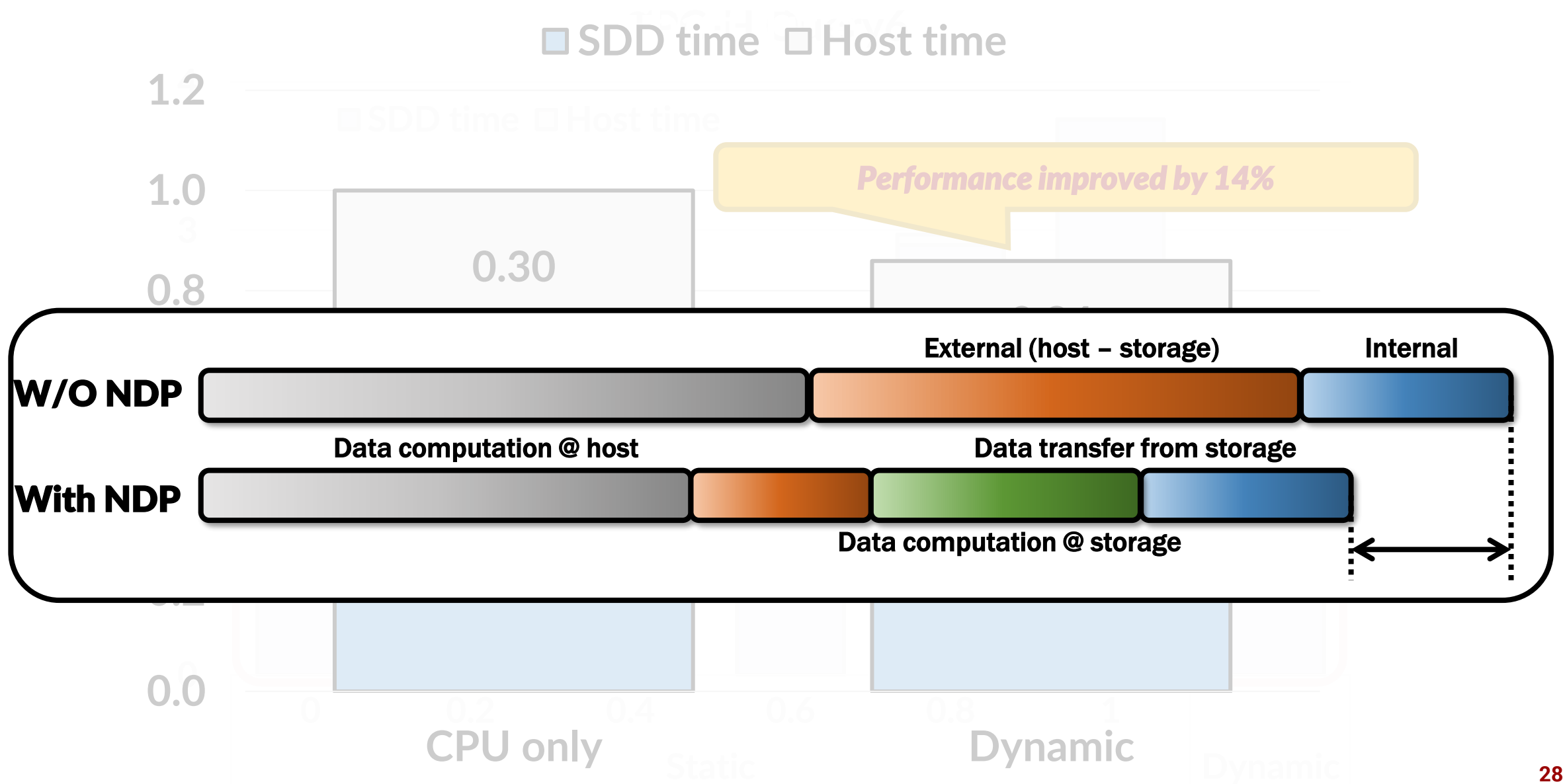




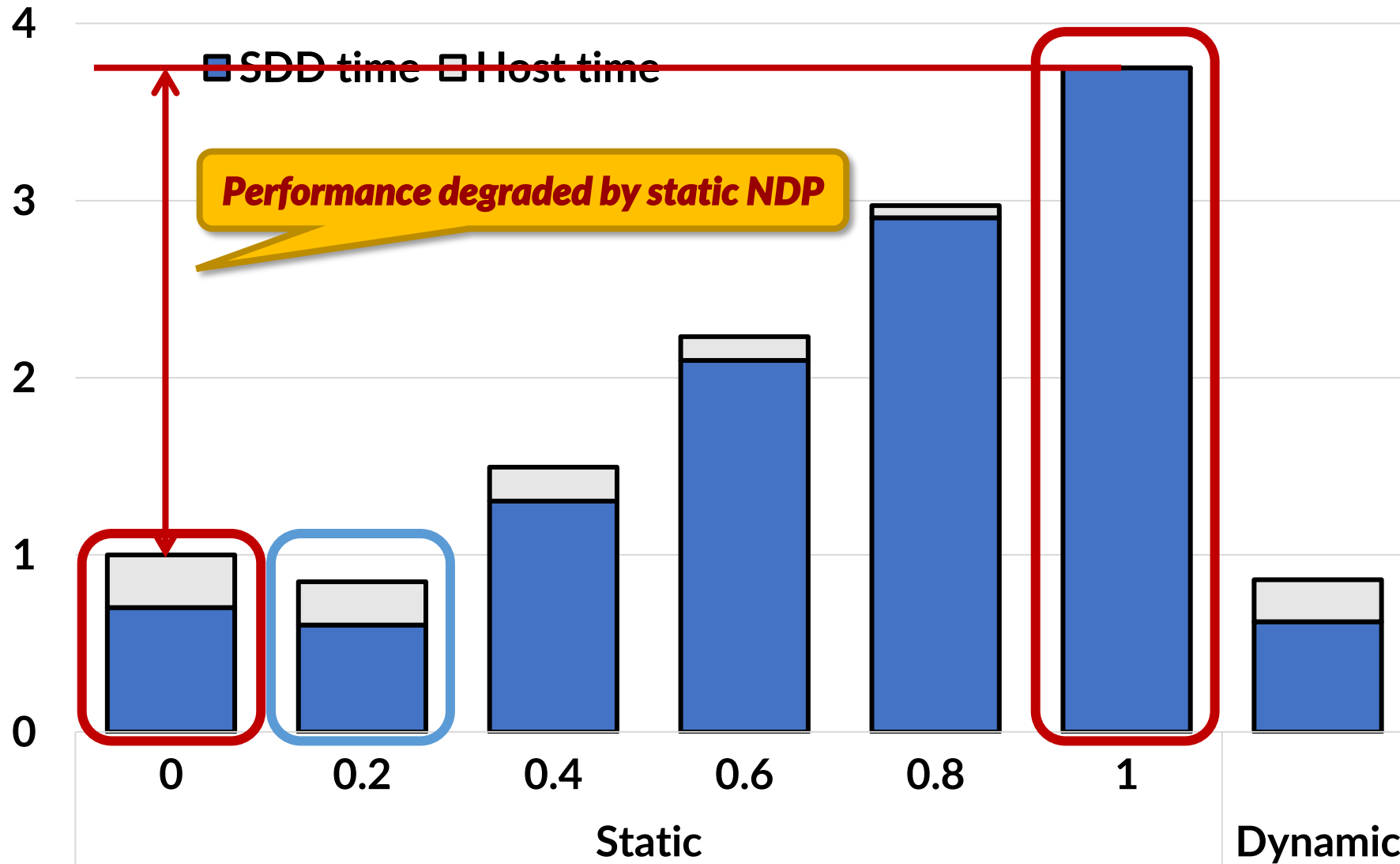
# Evaluation - Performance



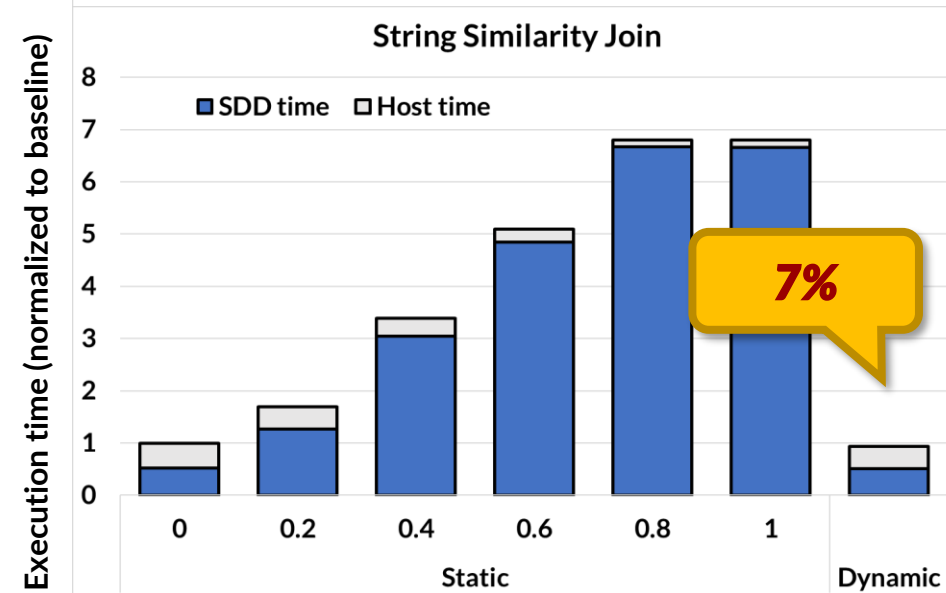
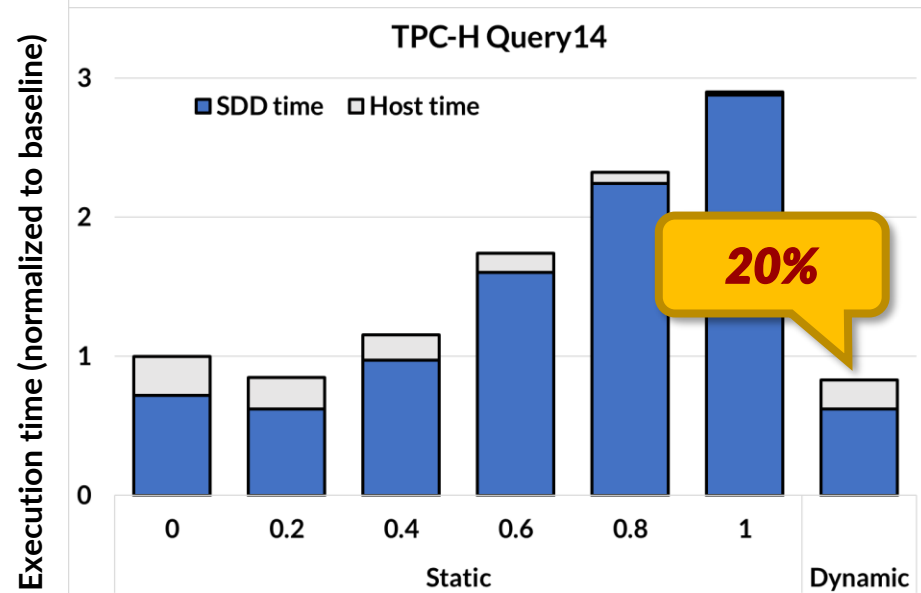
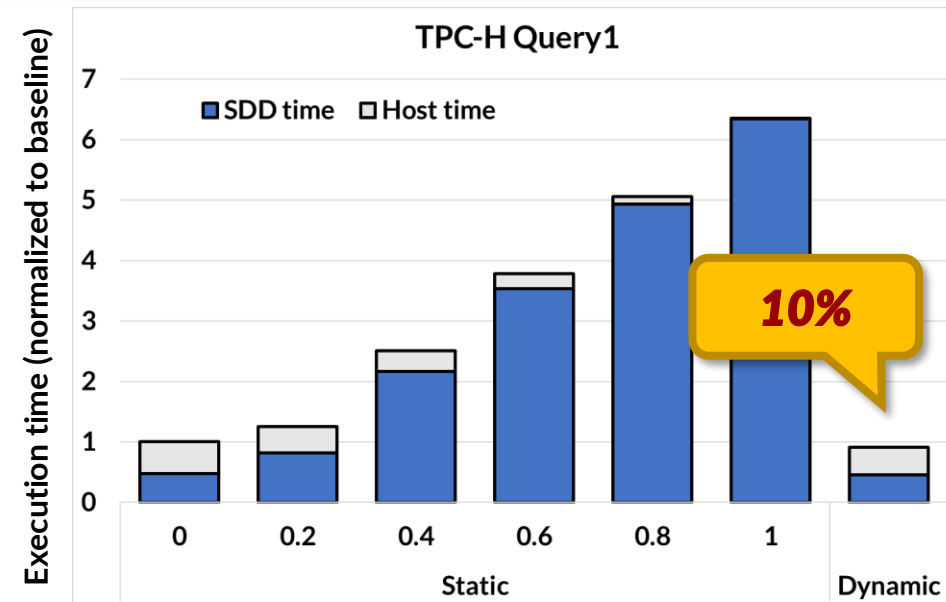
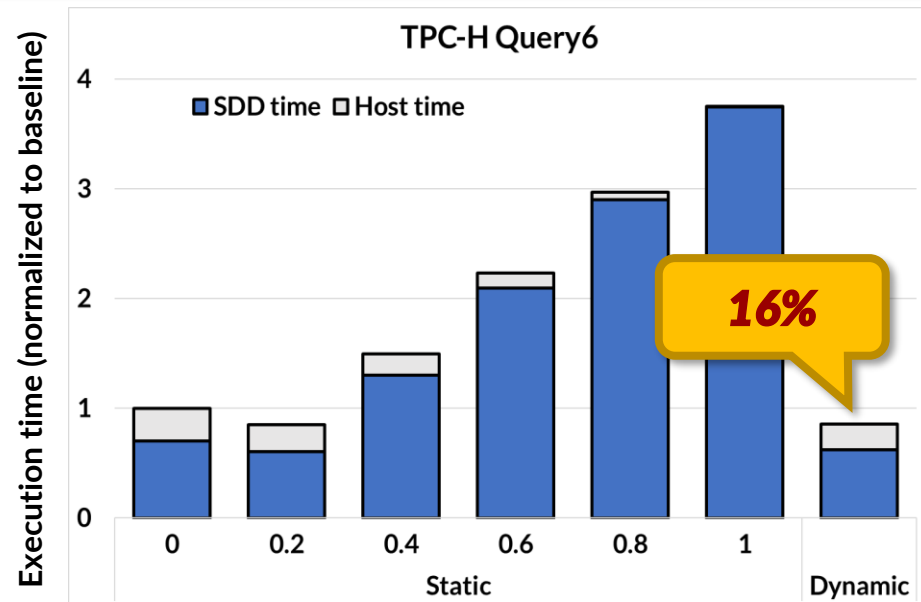
# Evaluation - Performance



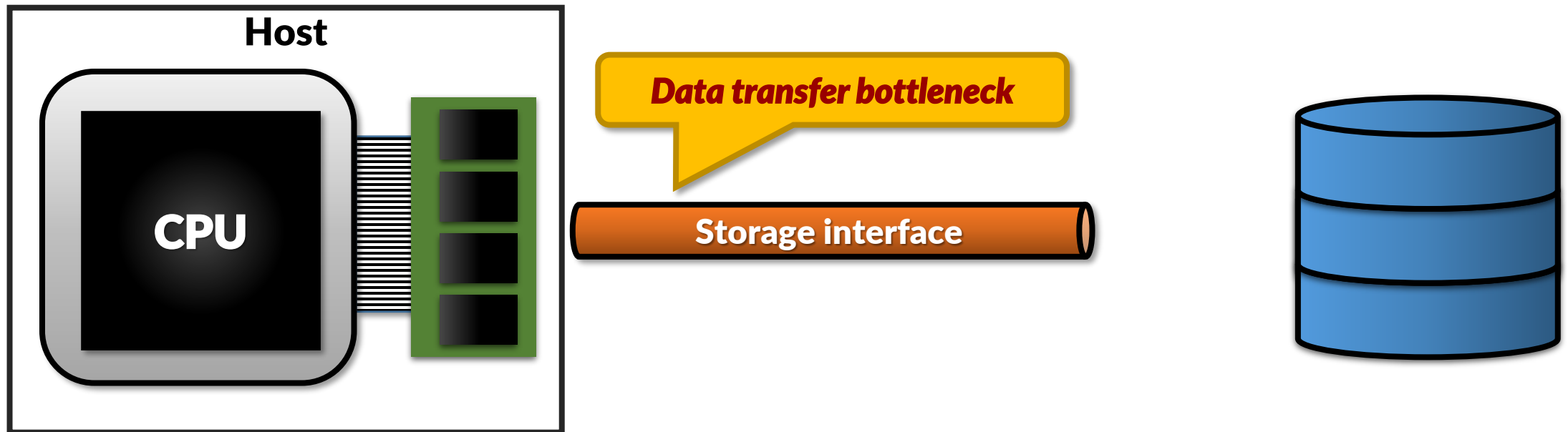
## TPC-H Query6



# Evaluation - Performance

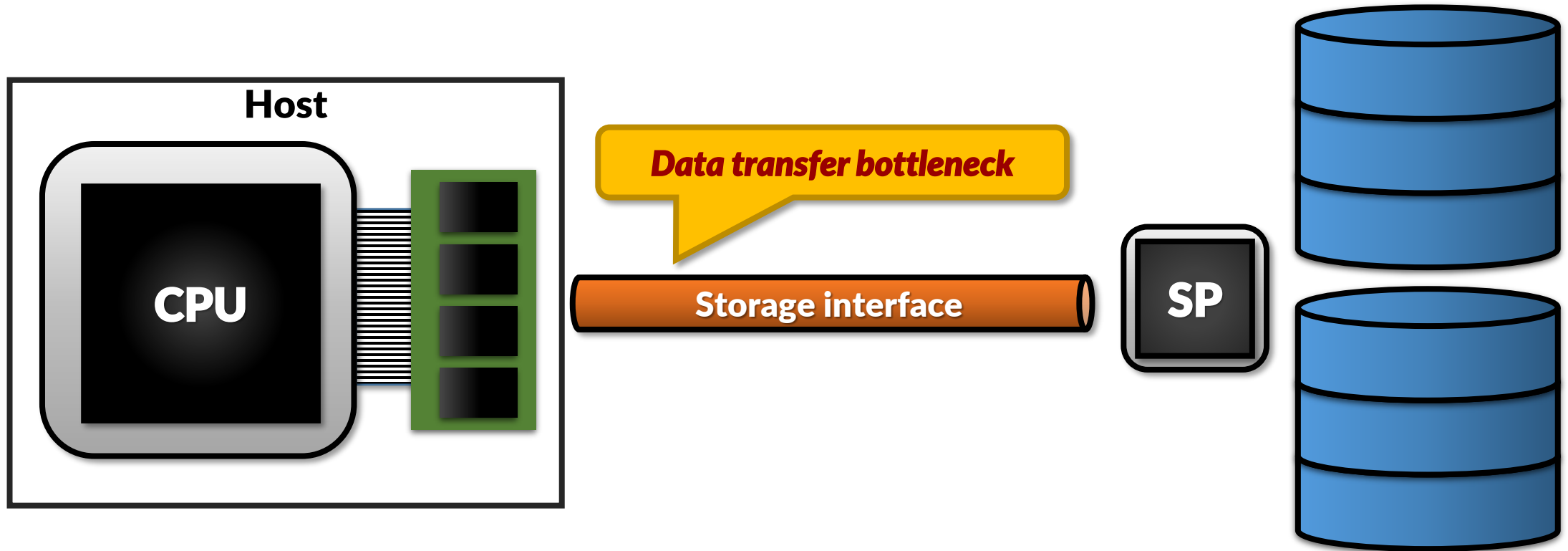


# Design Exploration – Higher Internal Bandwidth



**Commercial SSD maintains internal bandwidth  $\approx$  external bandwidth**

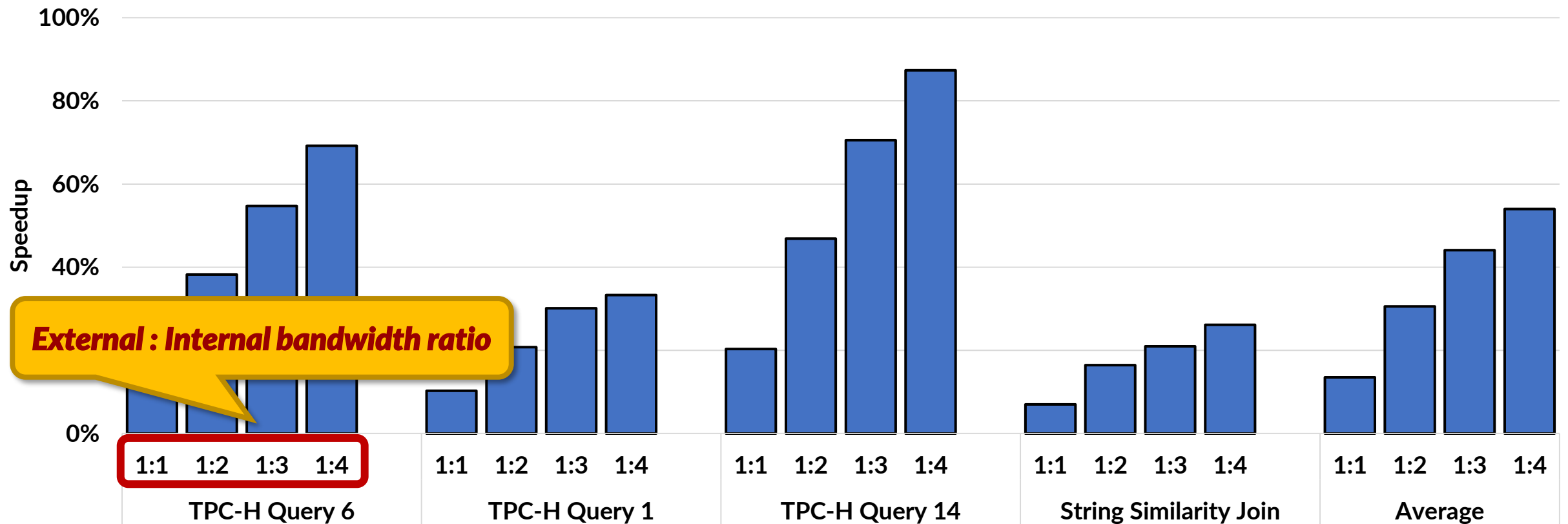
# Design Exploration – Higher Internal Bandwidth



***Higher internal bandwidth without increasing external bandwidth***

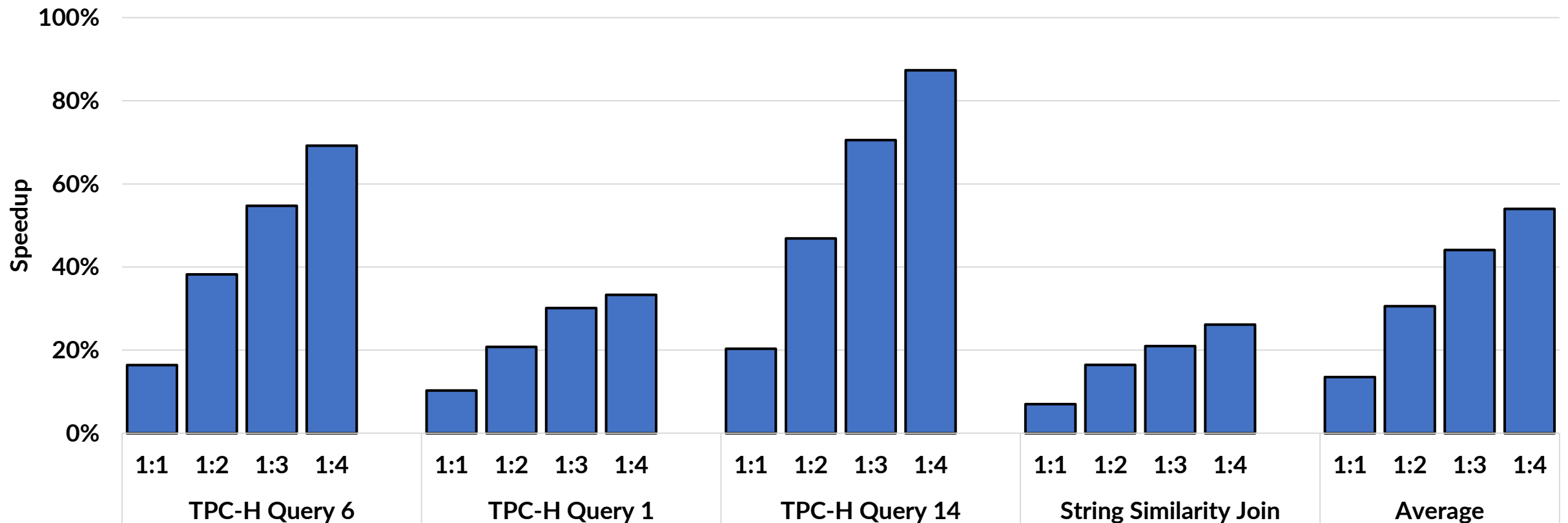


# Design Exploration – Higher Internal Bandwidth



# Design Exploration – Higher Internal Bandwidth

***Summarizer is effective if an SSD platform has higher internal bandwidth***

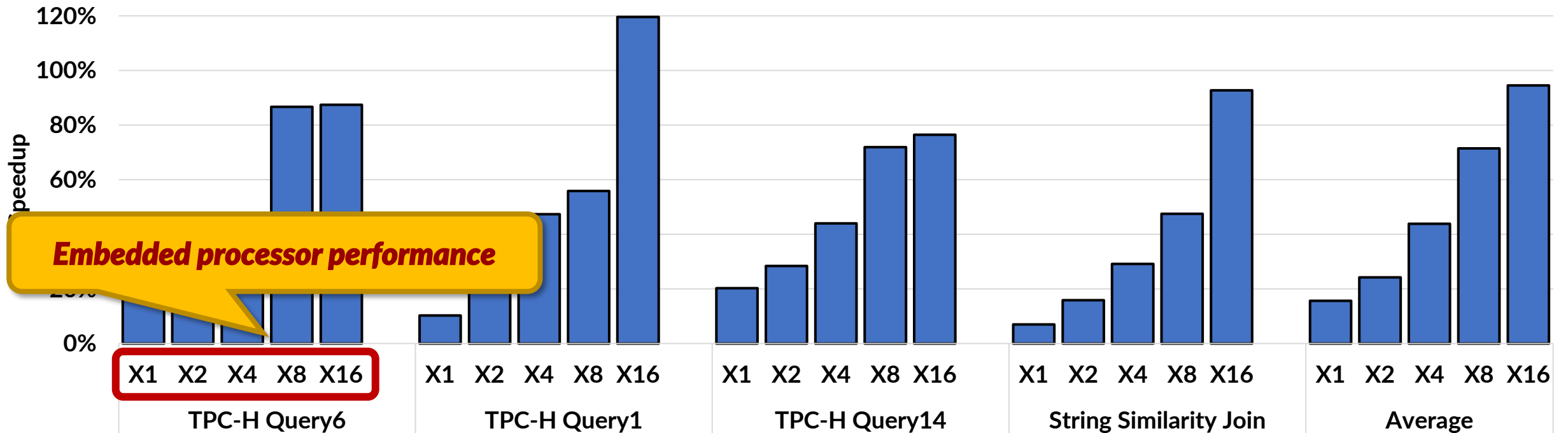


# Design Exploration – Better SSD Processor



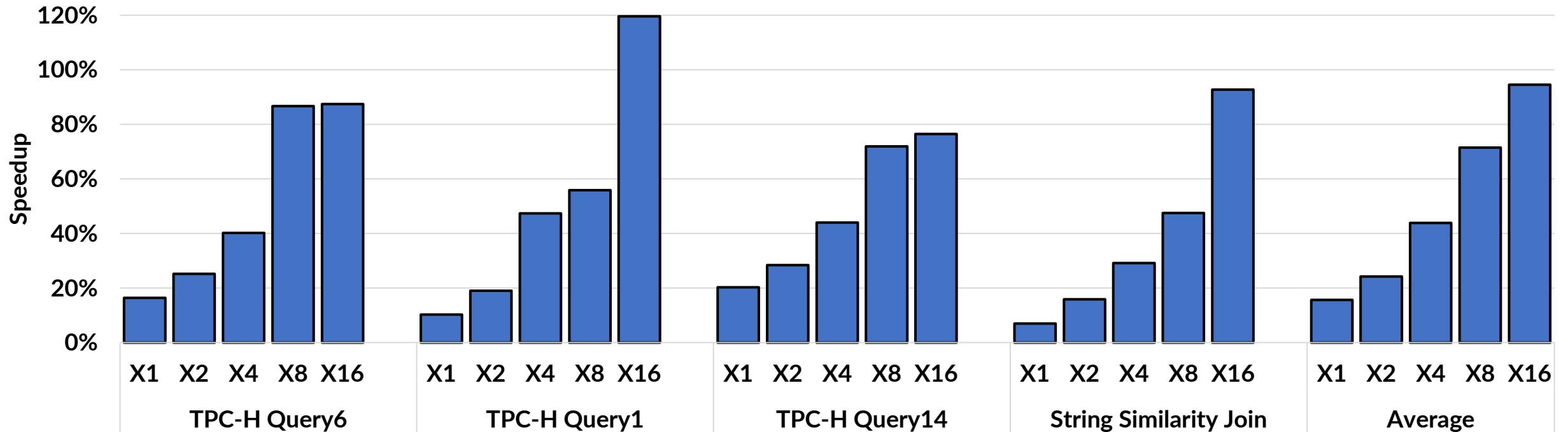
***Better embedded processor is cost effective***

# Design Exploration – Better SSD Processor



# Design Exploration – Better SSD Processor

**Summarizer is a cost effective NDP solution with powerful storage processors**



- ✓ ***Dynamic NDP framework for SSDs***
  - Opportunistically enables in-SSD processing
  - Page-level NDP control
  - Automatic workload partitioning
- ✓ ***Summarizer programming model***
  - Evaluation on the real development platform
  - Explored design space for future SSDs

*Thank you*

**Summarizer:**

**Trading Communication with Computing Near Storage**

**Gunjae Koo, Kiran Kumar Matam, Te I, H. V. Krishna Giri Nara, Jing Li,  
Hung-Wei Tseng, Steven Swanson, Murali Annavaram**

**(We thank to Dell EMC for supporting the SSD development board)**