Benefits of NVMe NVRAM vs NVDIMM, a database application example

Jerome Gaysse – IP-Maker

Goal

- OLTP database performance comparison with different storage options
  - Full flash SSD
  - NVMe NVRAM
  - NVDIMM

Methodology

- Part 1 - Existing hardware
  - Flash SSD, NVMe NVRAM, NVDIMM
  - MS SQL server 2016, HammerDB

- Part 2 - Estimation with new product design for higher capacity

OLTP performance

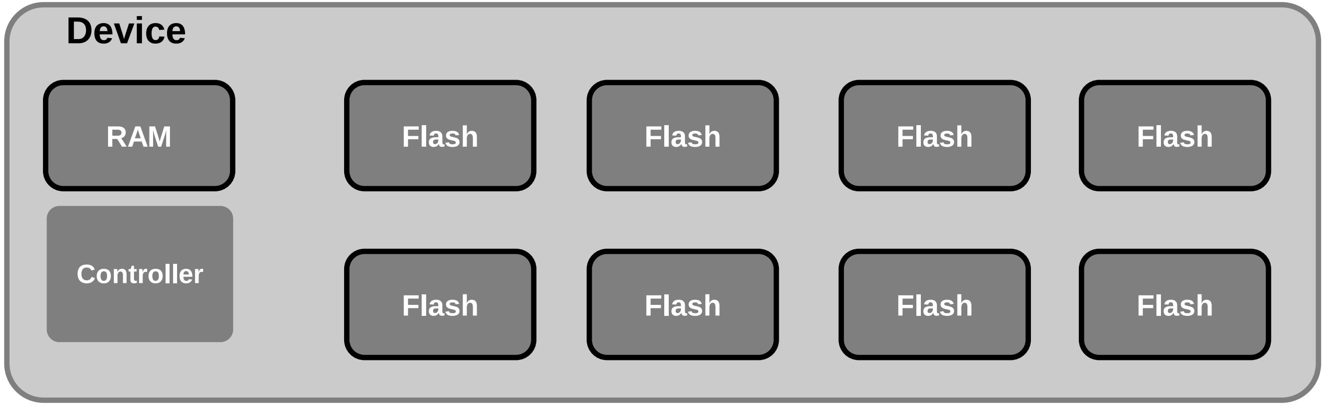
- A question of latency
- Many small read/write accesses to the DB file
- Write accesses to the LOG file

Part 1

- Flash SSD
  - Read 100µs latency
  - Write 500µs latency
- NVMe NVRAM
  - Read/write 12µs latency
- NVDIMM
  - Read/write 3µs latency

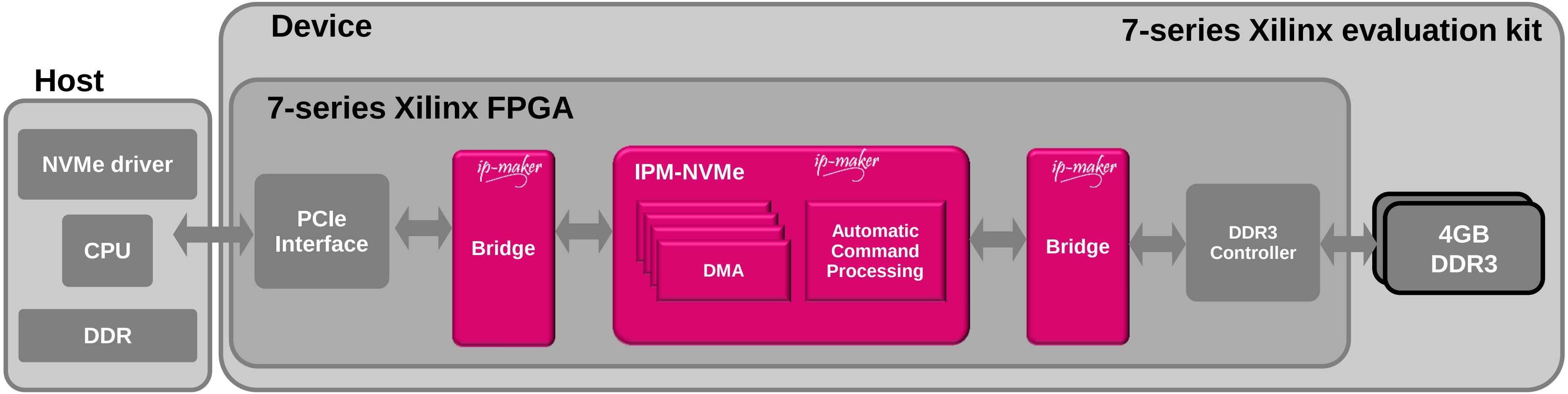
Flash SSD

- Latency : 100µs/500µs

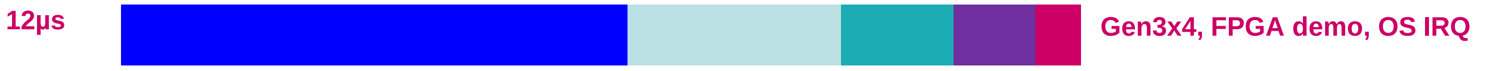


NVMe RAM

- 12µs latency



Latency details



🟥 NVMe IP (Command fetch + data management): 0,6µs (clock 125MHz)

🟦 File system+ NVMe driver: 2,85

🟩 Doorbell, Command read: 1.5µs

🟪 Data transfer: 1.1µs (Gen3 speed, 4 lanes)

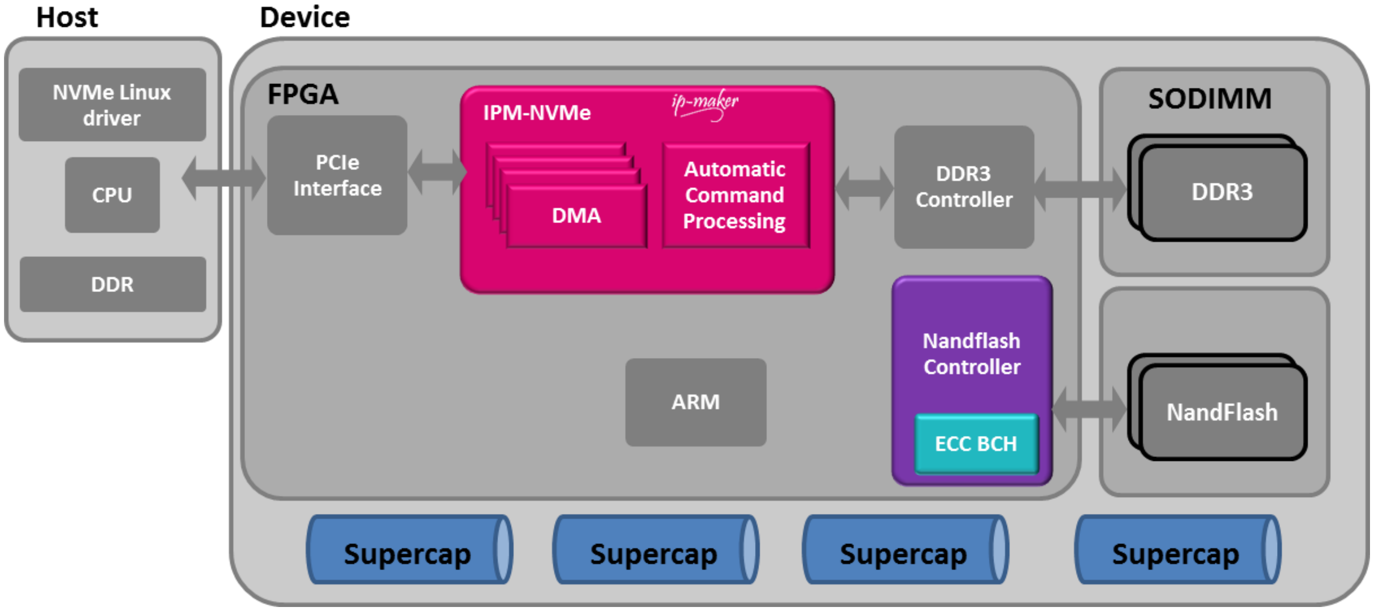
🟦 Host IRQ management + PCIe latency: 7.7

Options to reduce latency

- PCIe gen 4
- Command Memory Buffer (CMB)
- Command Memory Buffer (CMB) with persistent memory
- Polling mode

- => from 12 to 5µs

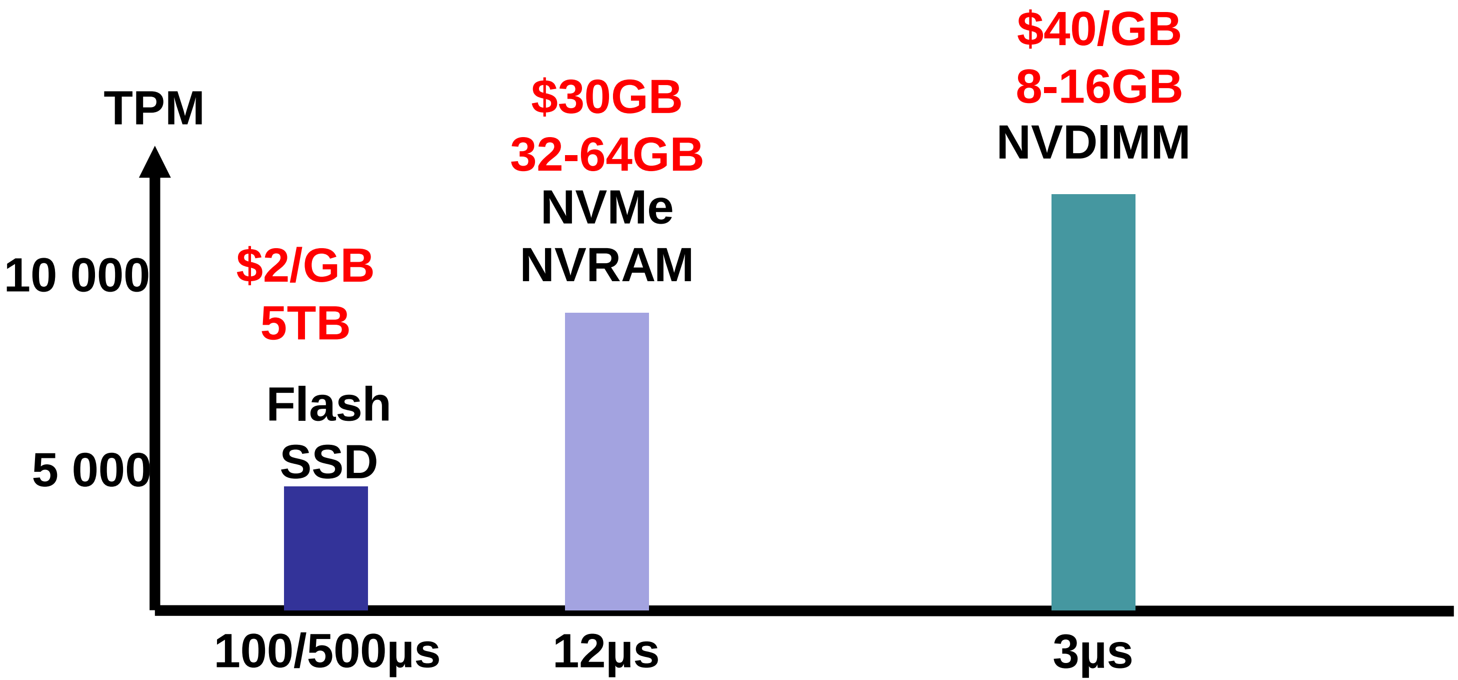
NVMe NVRAM Implementation (NVDIMM-N like)



NVDIMM

- NVDIMM simulation using:
  - 4GB LRDIMM
  - RAM disk software
- Latency measured with FIO: 3µs

Performance results



The price for performance

- Flash: $2/GB, 5TB
  - 4K TPM
- NVMe NVRAM: $30/GB, 32GB
  - 10K TPM
- NVDIMM: $40/GB, 8GB
  - 14K TPM

- What about TCO for TB database?

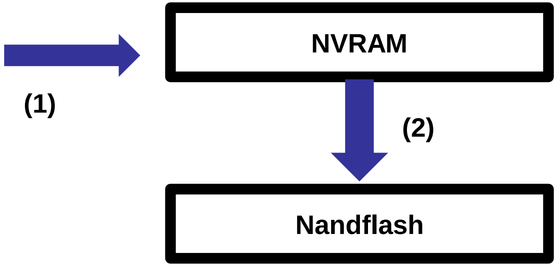
Part 2

- NVMe NVRAM
  - High storage capacity ?

- NVDIMM
  - High storage capacity ?

NVMe NVRAM Product design

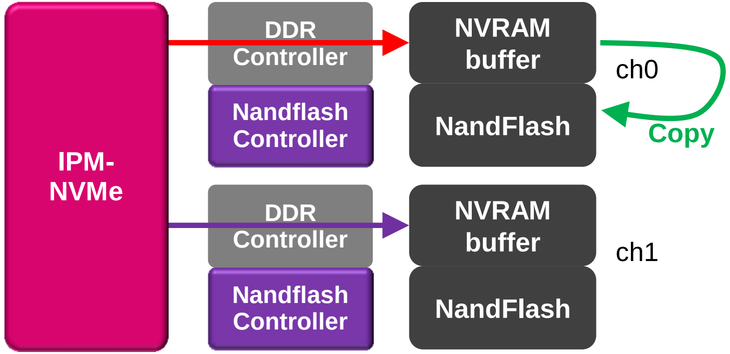
- Achieving high capacity and low write latency
  - Non-volatile buffer for low latency
  - Nandflash storage for high capacity
  - Highly parallel implementation for high throughput
- Based on pairs of NVRAM and nandflash memories.
  - The data is first coming from the controller (1).
  - Then it is copied in the nandflash (2).



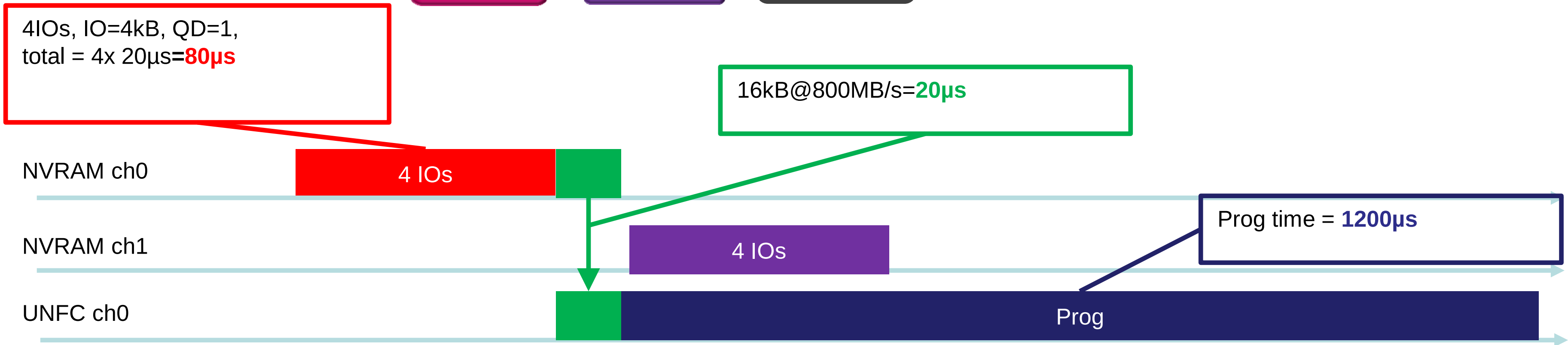
Theory of operation (1/2)

- The first 4 IOs are sent to the NVRAM buffer 0.
- The second 4 IOs are sent to the NVRAM buffer 1.
- During this time, the data is read from the NVRAM buffer 0 and written into the nandflash channel 0.

Theory of operation (2/2)



Sequence:
1=write to NVRAM CH0
2=write to NVRAM CH1
3=copy from NVRAM to Flash
4=prog flash
…

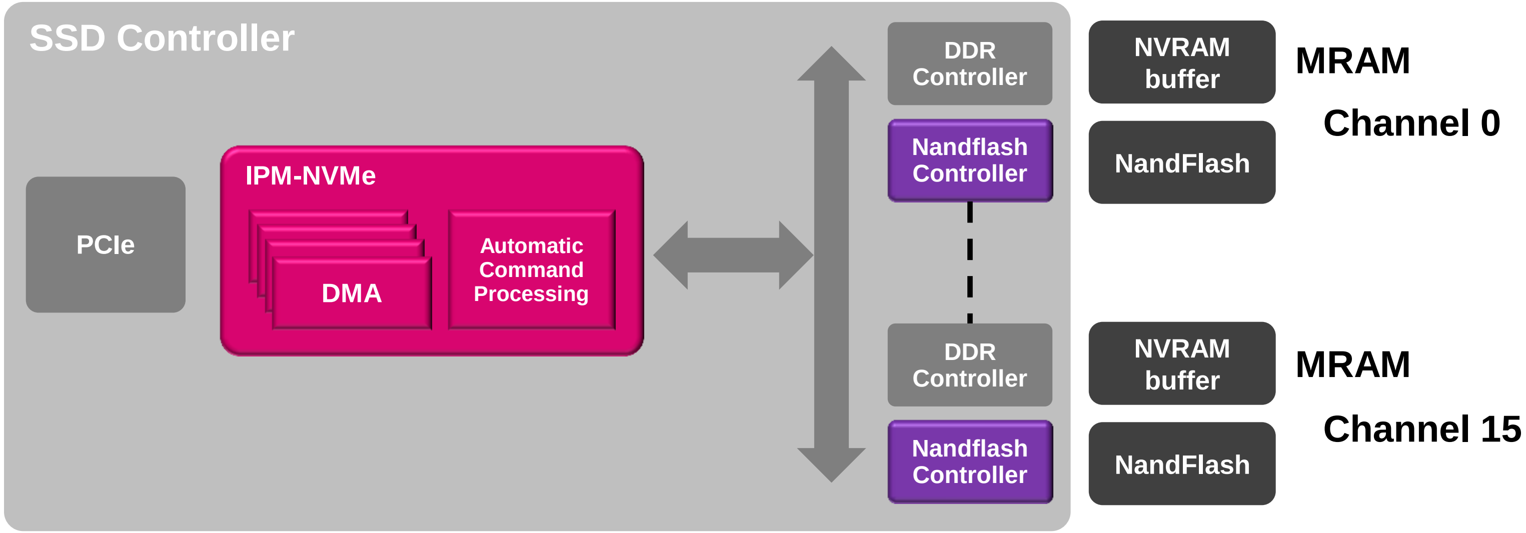


4IOs, IO=4kB, QD=1,
total = 4x 20µs=**80µs**

16kB@800MB/s=**20µs**

Prog time = **1200µs**

NVMe NVRAM Implementation (with MRAM)

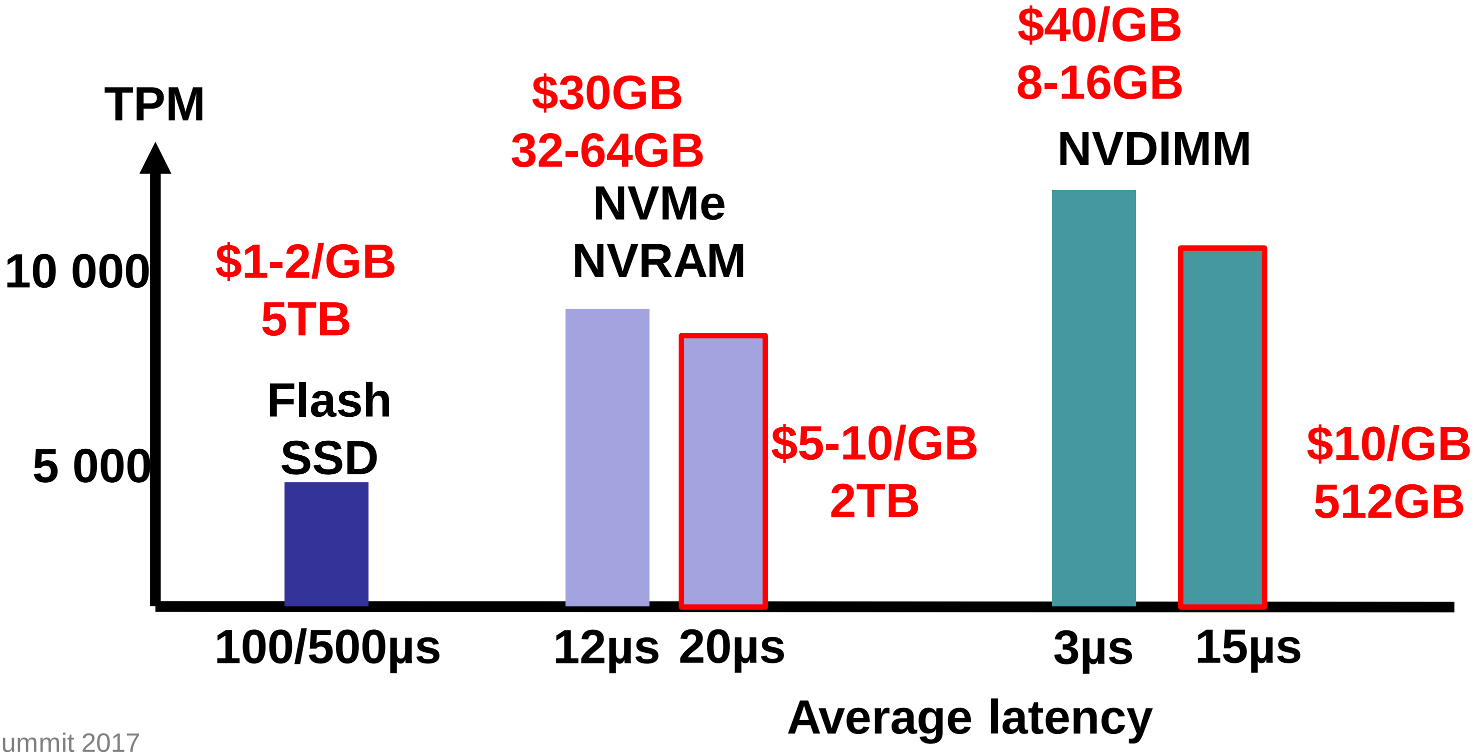


20µs latency

NVDIMM

- Higher storage capacity?
  - Yes, few hundreds of GB of Flash can be added
- Highly parallel design?
  - No, limited by PCB area
  - Average latency to increase

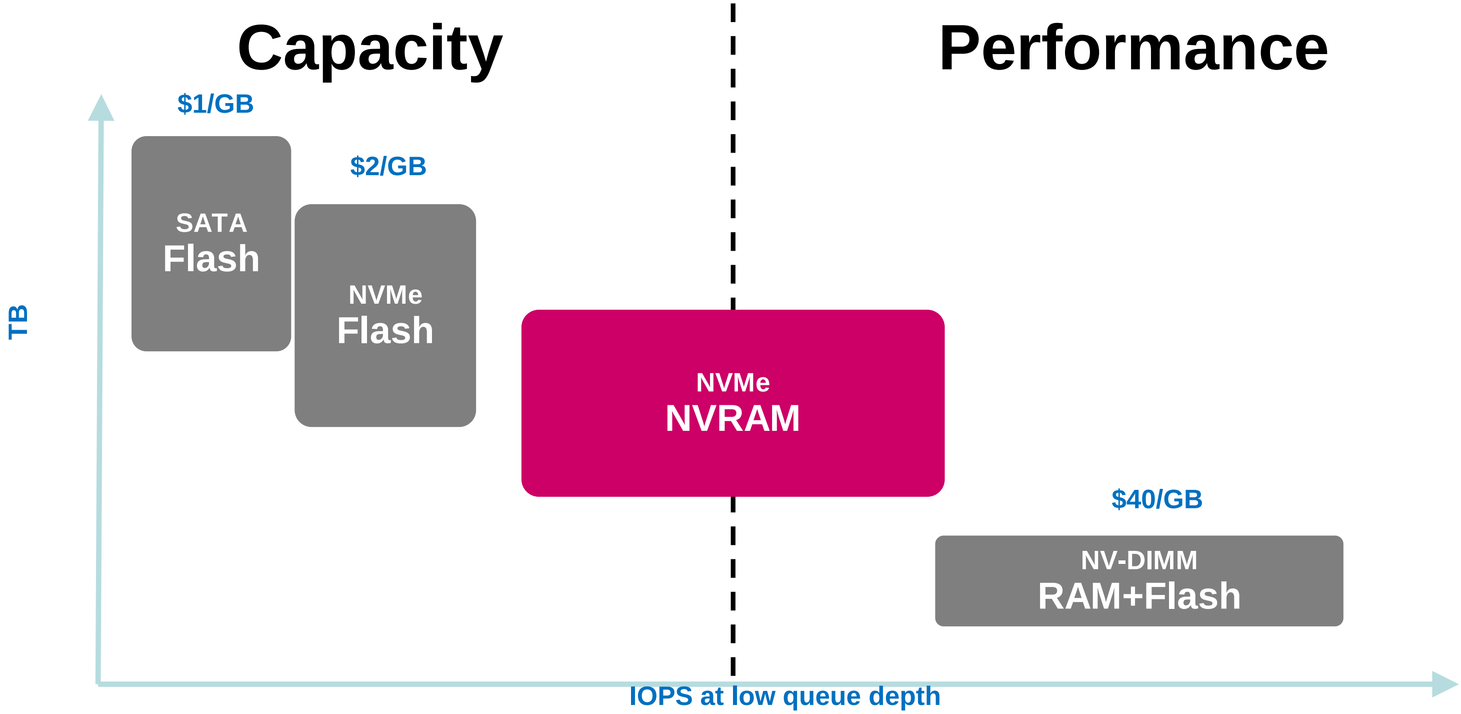
Performance estimation



The price for performance

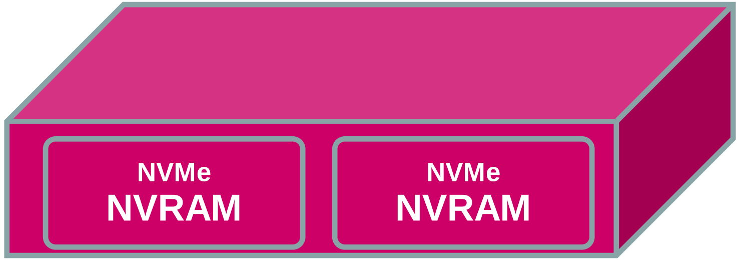
- Flash: $2/GB, 5TB
  - 4K TPM
- NVMe NVRAM: $5/GB, 2TB
  - 9K TPM
- NVDIMM: $10/GB, 512GB
  - 13K TPM

Synthesis


Capacity
Performance
$1/GB
SATA Flash
$2/GB
NVMe Flash
NVMe NVRAM
$40/GB
NV-DIMM RAM+Flash
TB
IOPS at low queue depth

NVMe NVRAM vs NVDIMM

OLTP application



Log DB

NVMe NVRAM:
for both Logs and DB files

Jerome Gaysse

jerome.gaysse@ip-maker.com

www.ip-maker.com