Query Processing on Smart SSDs

Opportunities and Challenges

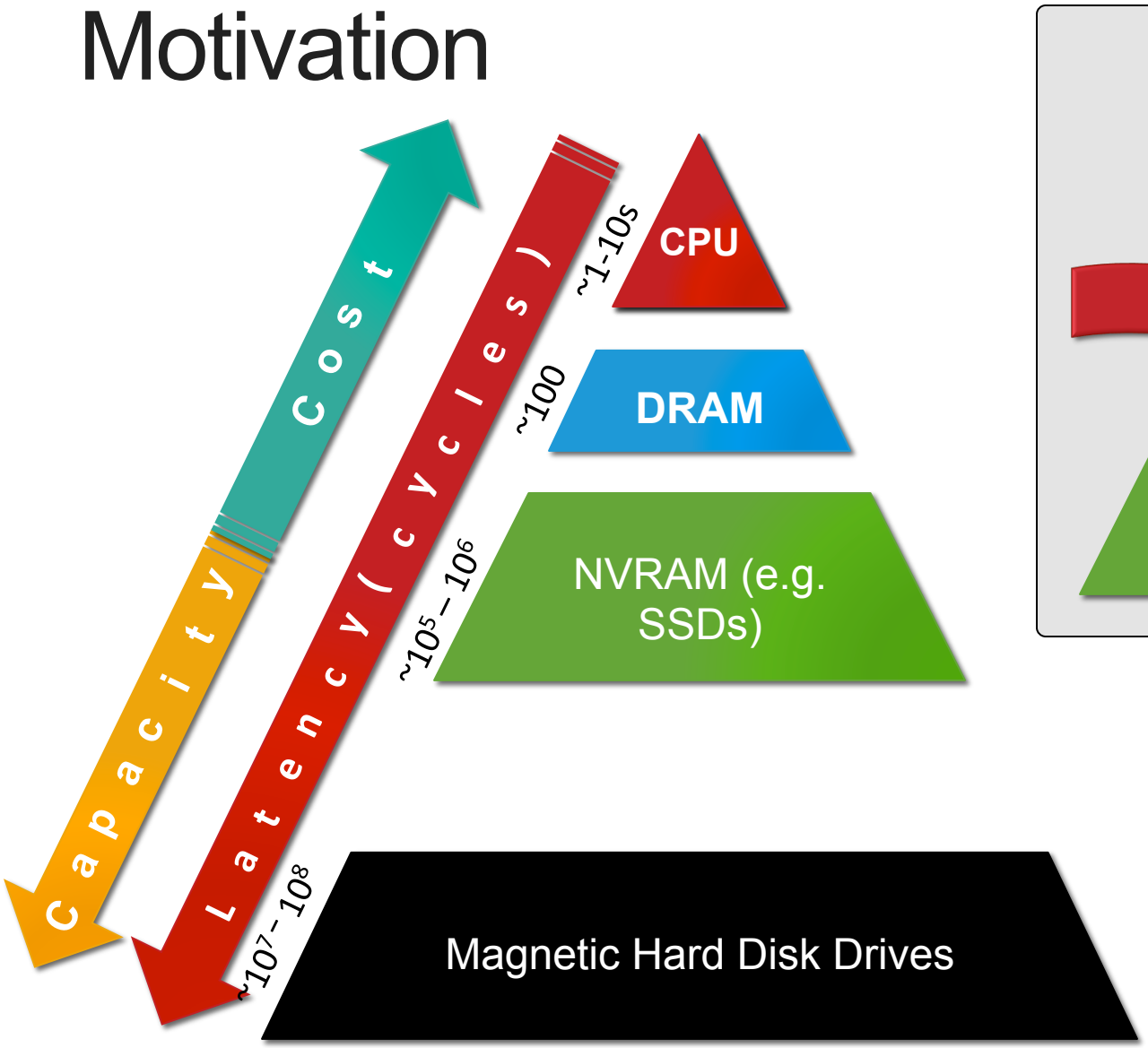
Jaeyoung Do[1], Yang Seok Ki[2], Jignesh M. Patel[1], Chanik Park[2], Kwanghyun Park[1], David J. DeWitt[3]

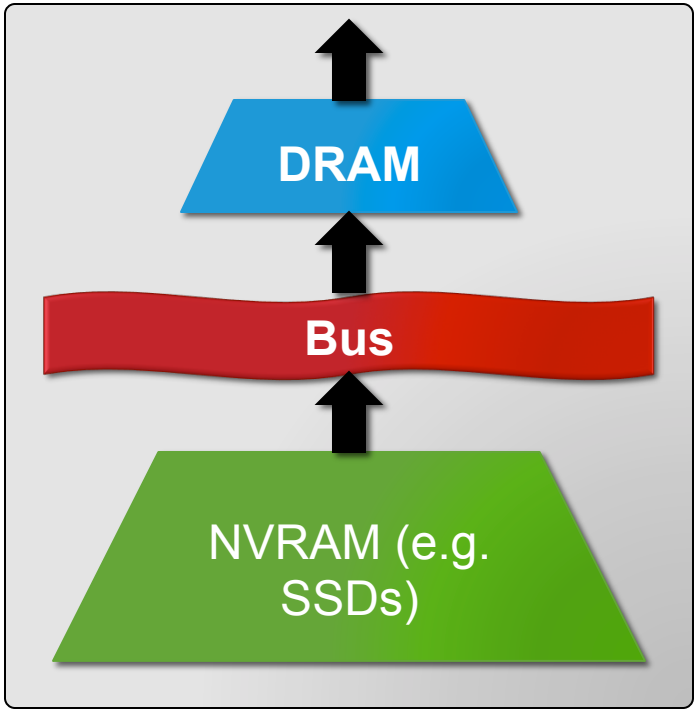
[1] University of Wisconsin – Madison
[2] Samsung Semiconductor Inc.
[3] Microsoft Jim Gray Systems Lab

Contact: jignesh@cs.wisc.edu

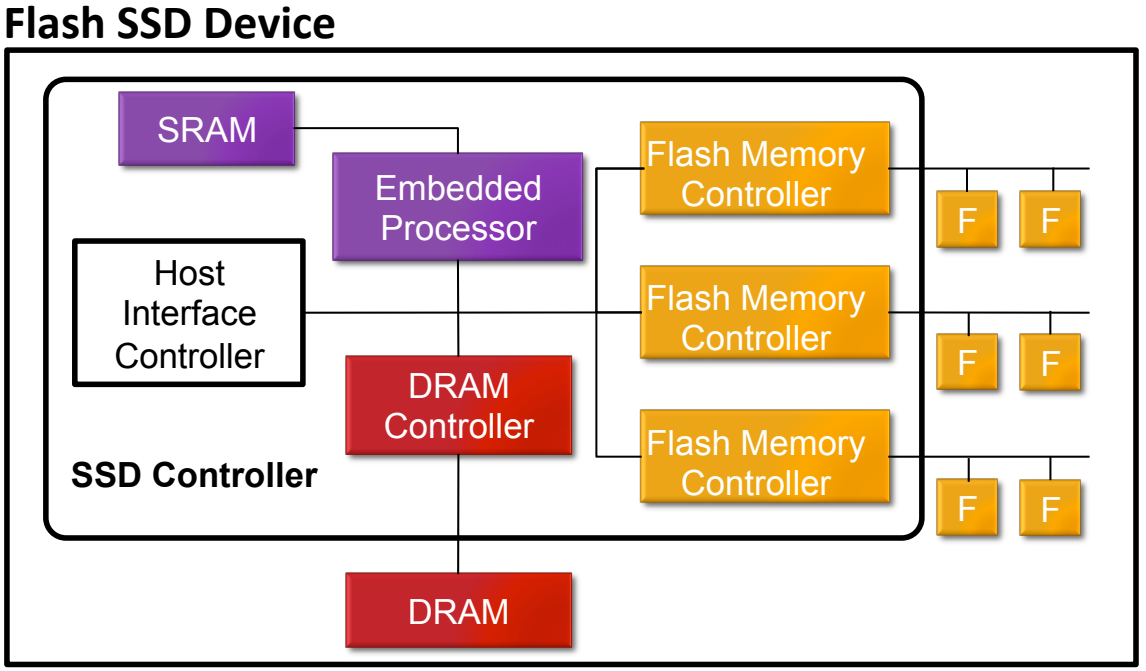
The opinions expressed here are those of the authors, and not necessarily of the organizations to which the authors belong.

Motivation


Cost
Capacity
Latency (cycles)
~1-10s
CPU
~100
DRAM
~10^5 – 10^6
NVRAM (e.g. SSDs)
~10^7 – 10^8
Magnetic Hard Disk Drives

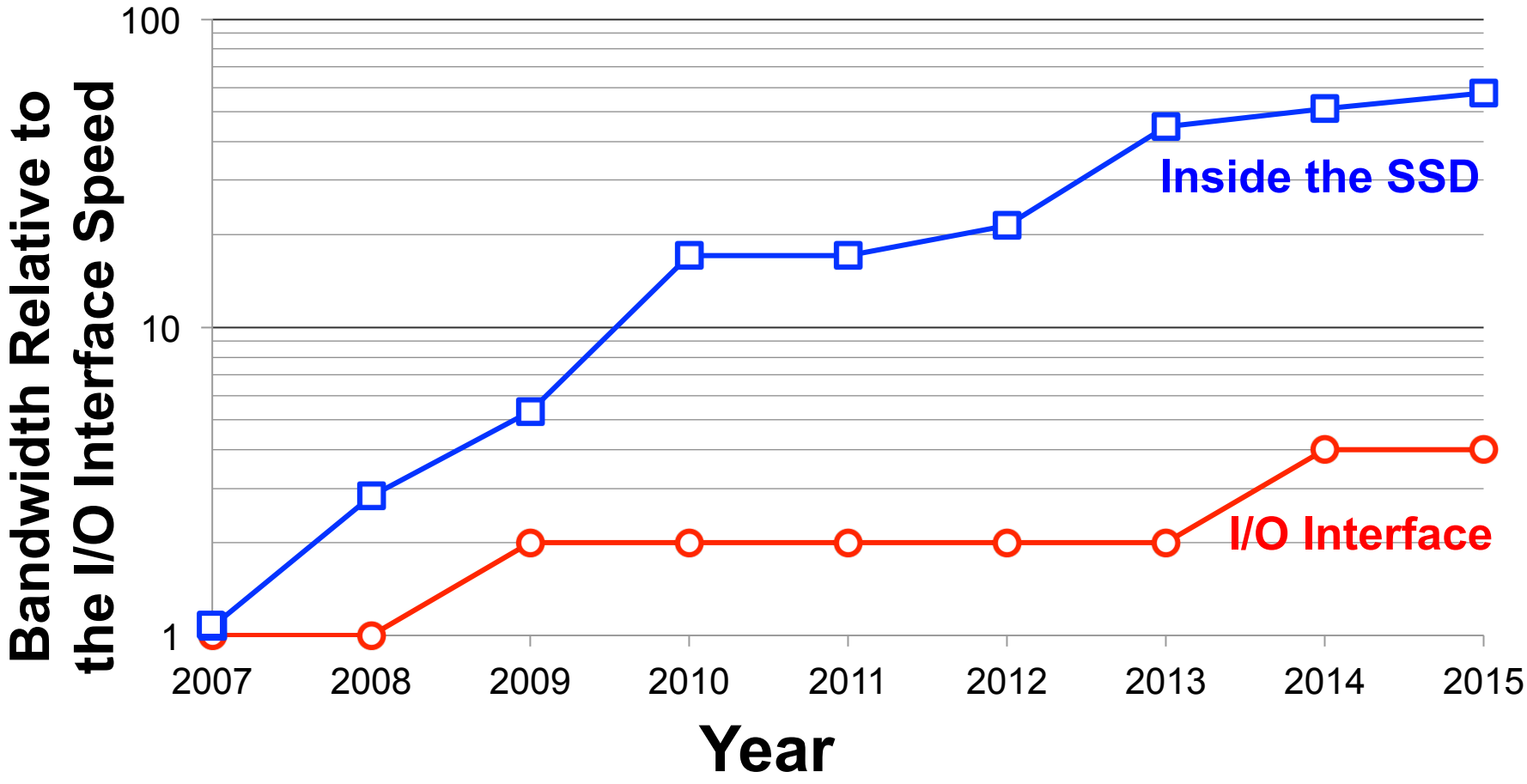

DRAM
Bus
NVRAM (e.g. SSDs)

Inside a Modern SSD



CPU	• Embedded processors (low power)
Memory	• General-purpose memory
I/O	• High I/O bandwidth inside the flash chips

Situation today: Drinking data through a narrow straw



The Opportunity

- Push code through the "straw," thereby drinking smaller amounts through the straw

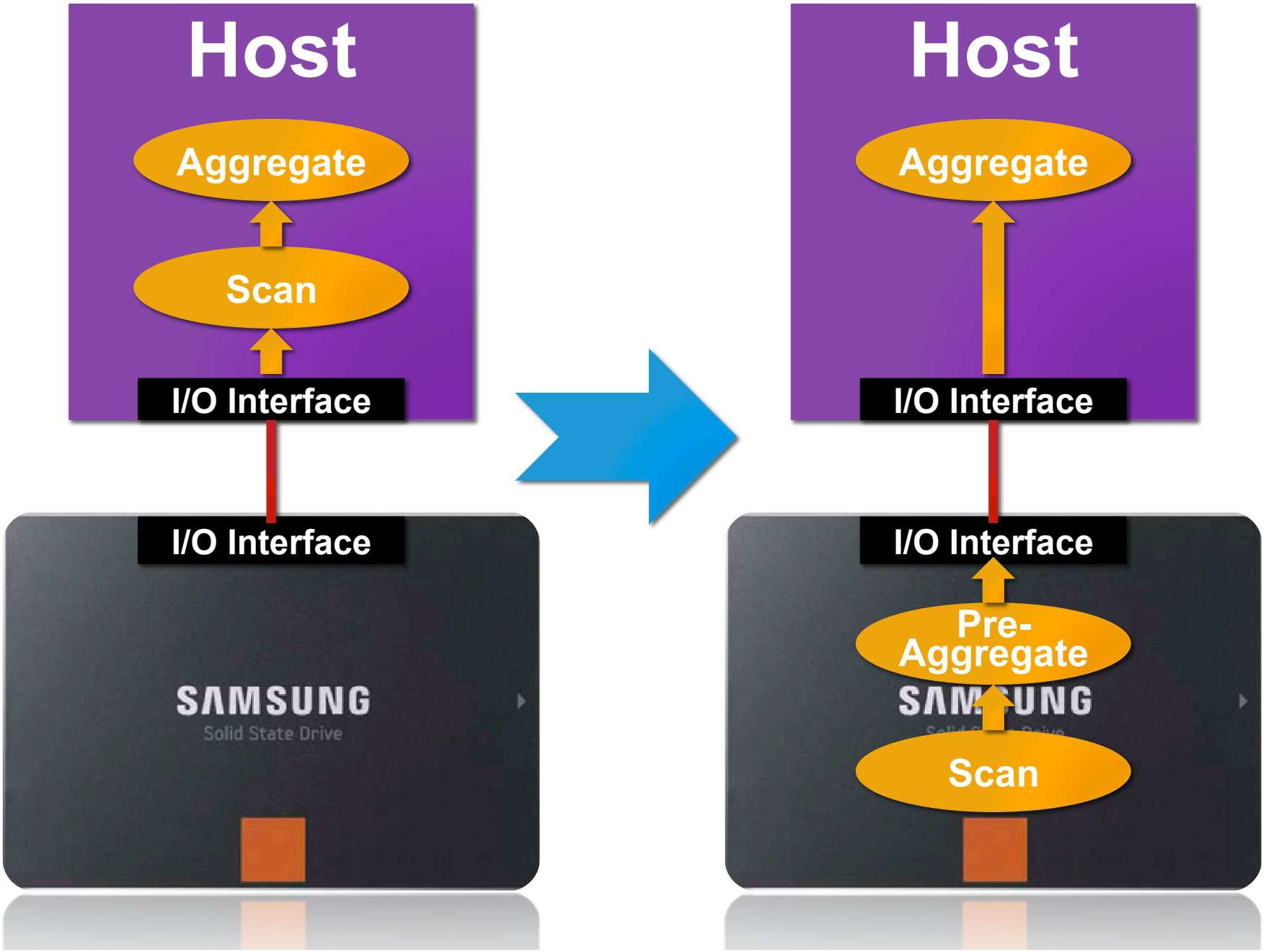
  Philosophically similar to what Netezza and Exadata do today, but using "commodity" resources inside the SSD

Potential Advantages

- Higher performance
- Lower energy consumption (Joules/query)
- Lower cost when building balanced systems
- Simpler appliances

Current Challenges

- Programming tools for the SSD are primitive
- Not enough "spare" processing power
- Some hardware architectural limitations
- Deeper DBMS implications: query optimization, buffer pool caching, updates, transaction mgmt., etc.


Host
Aggregate
Scan
I/O Interface
I/O Interface
SAMSUNG
Solid State Drive
Host
Aggregate
I/O Interface
I/O Interface
Pre-
Aggregate
Scan

Implementation Details

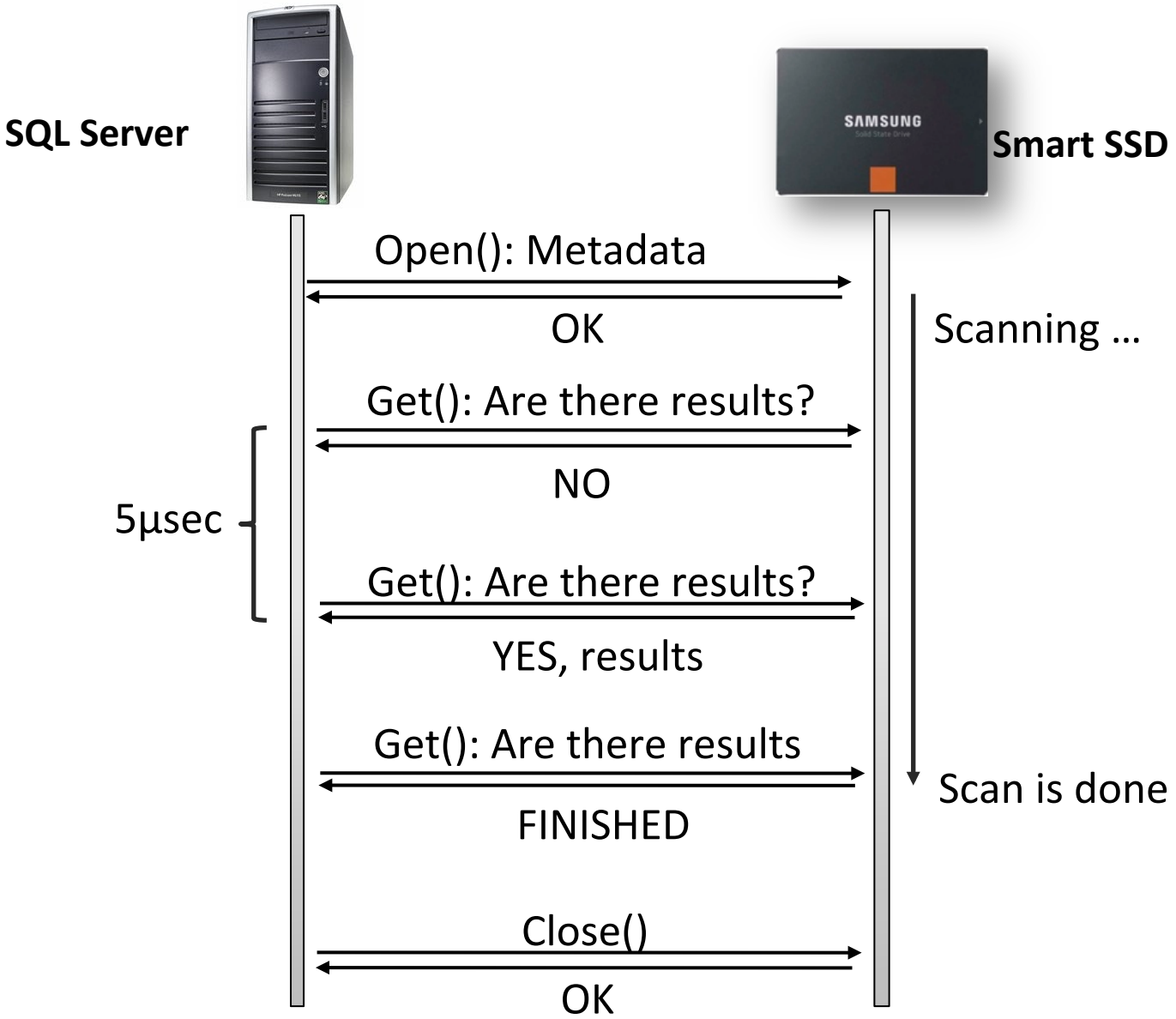
Samsung Smart SSD

- **New Smart SSD APIs** to run database operations inside the SSD
- Open(), Get(), Close(), SendToHost()

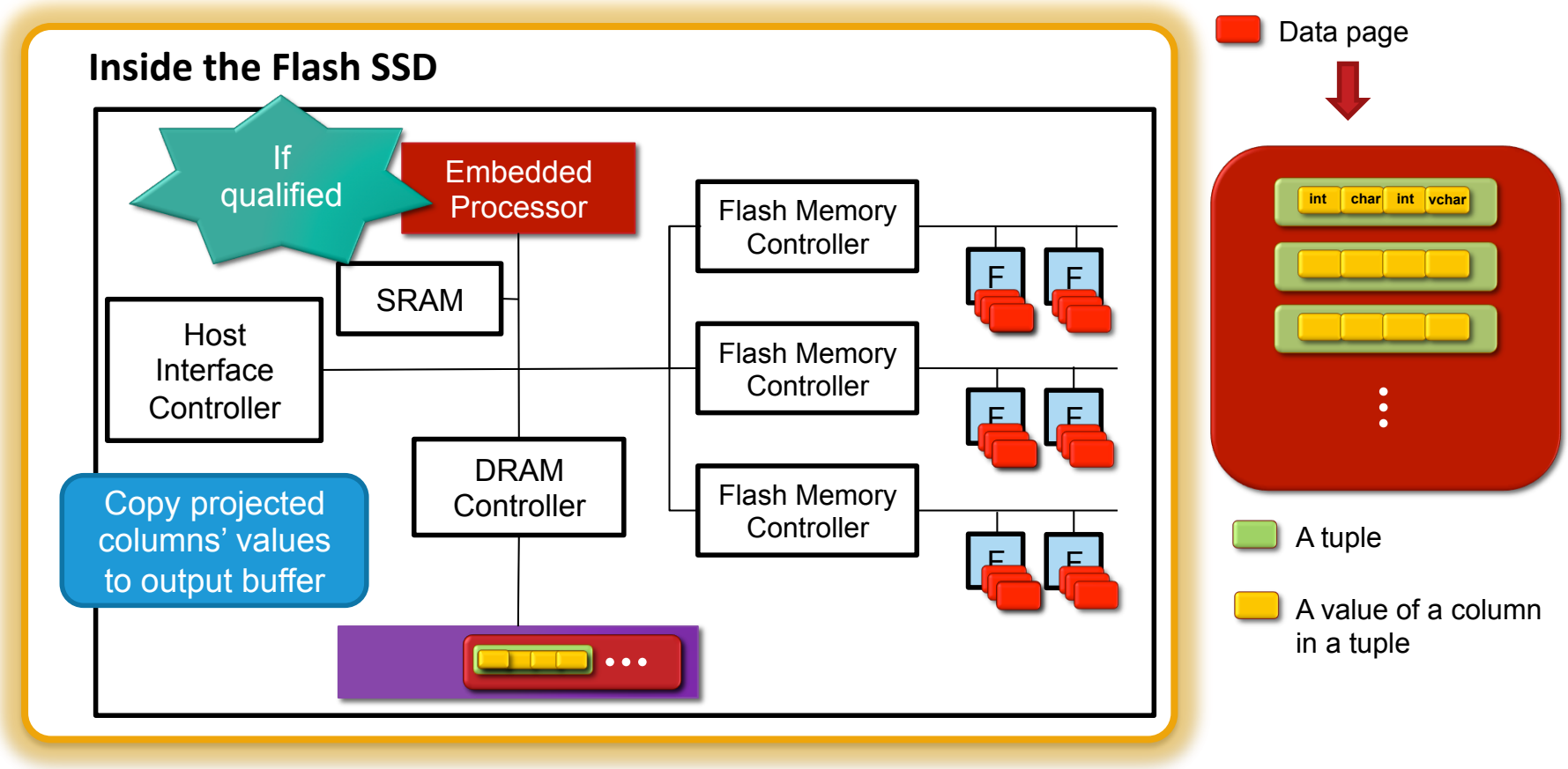
SQL Server

- Modified the query processor to hard-code selected Smart SSD plans
- Implement Smart SSD executors

Communication Flow – e.g. Scan



Data Flow inside the SSD (NSM format)



Storage Formats

- NSM layout

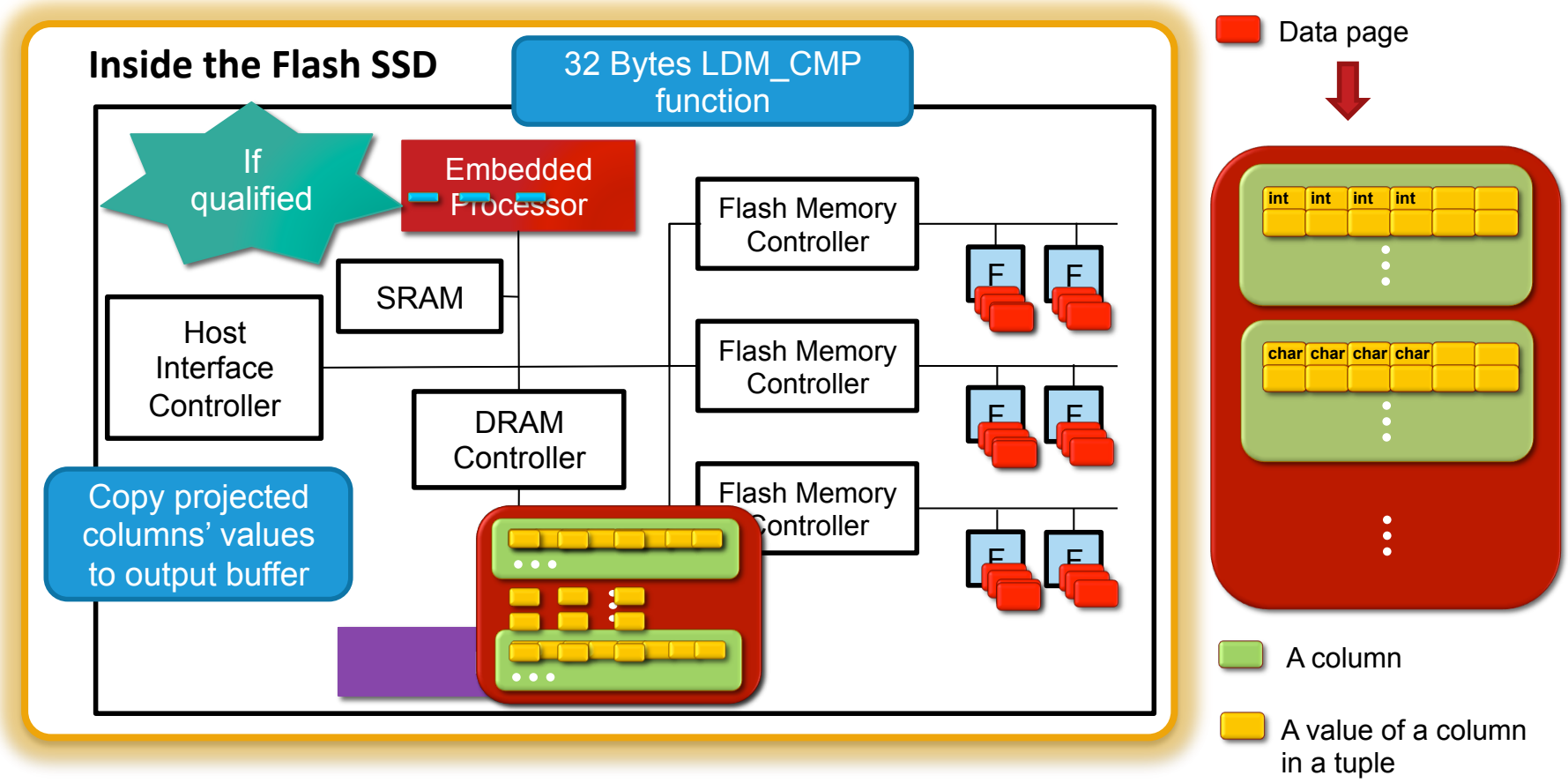
Data Page

Tuple

Column

- PAX layout

Data Flow inside the SSD with the PAX layout



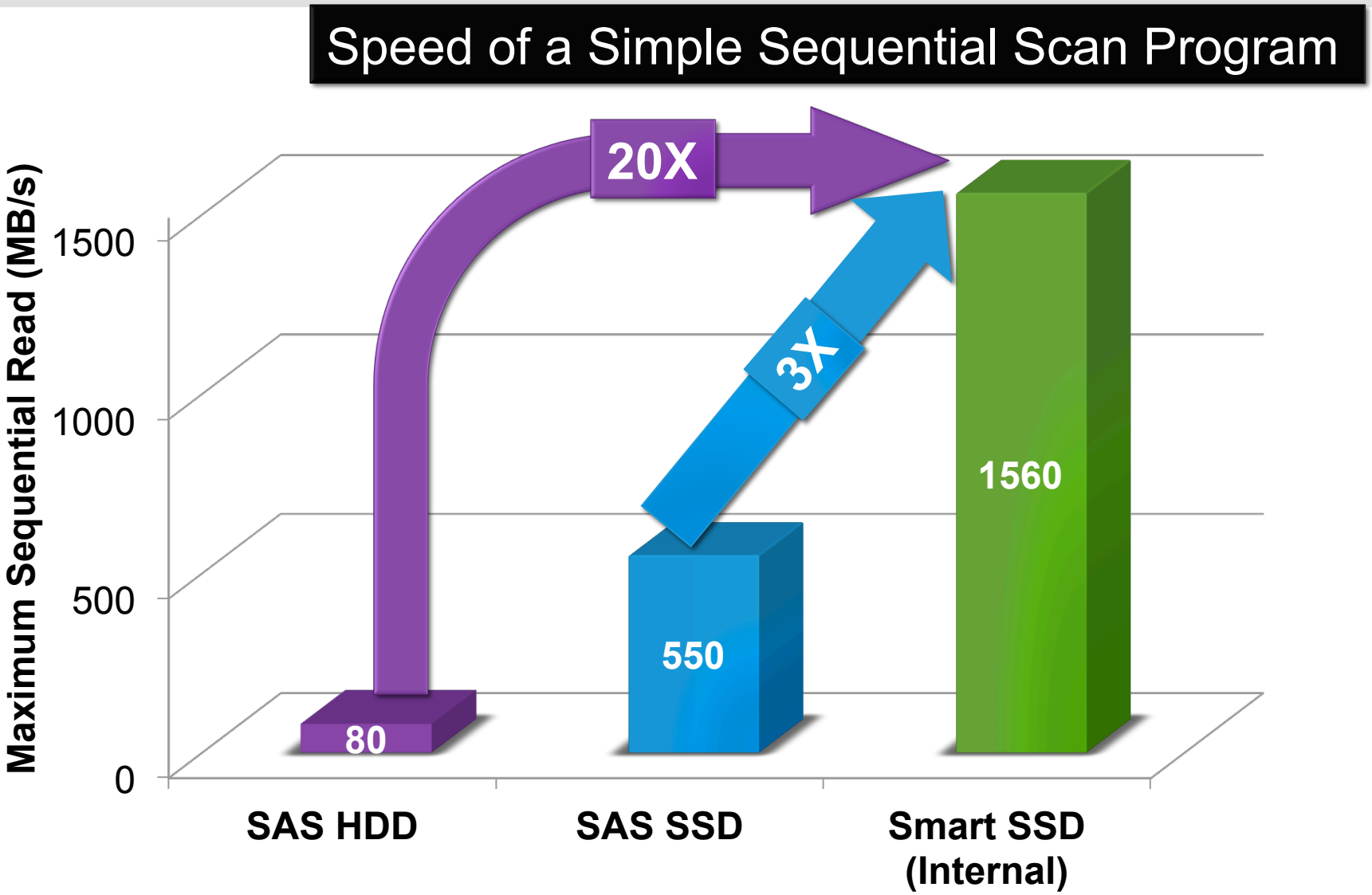
System Configuration

SQL server (Host machine)
64-bit Windows 7
32GB DRAM (24GB dedicated to the DBMS)
2.66 GHz Dual Quad Core Intel Xeon5430 32KB L1 cache, 6MB L2 cache
Two 7.5 RPM SATA HDDs one for the OS and one for the Log
LSI four-port SATA/SAS 6Gbps HBA (host bus adapter)

Flash SSD
Samsung 400GB SAS SSD
Two ARM Cores
16KB SRAM for each core
256KB DRAM for each core
1MB DRAM output buffer

HDD
HP 146GB 10K RPM SAS HDD

Upper Bound on Expected Performance Gains



Experimental Setup: Synthetic Data

Each table has a number of integer-typed columns

Table	# Columns	Table Size (GB)	# tuples (millions)	# tuples/ page
Synthetic 4	4	10	400	323
Synthetic16	16	30	400	109
Synthetic64	64	110	400	29

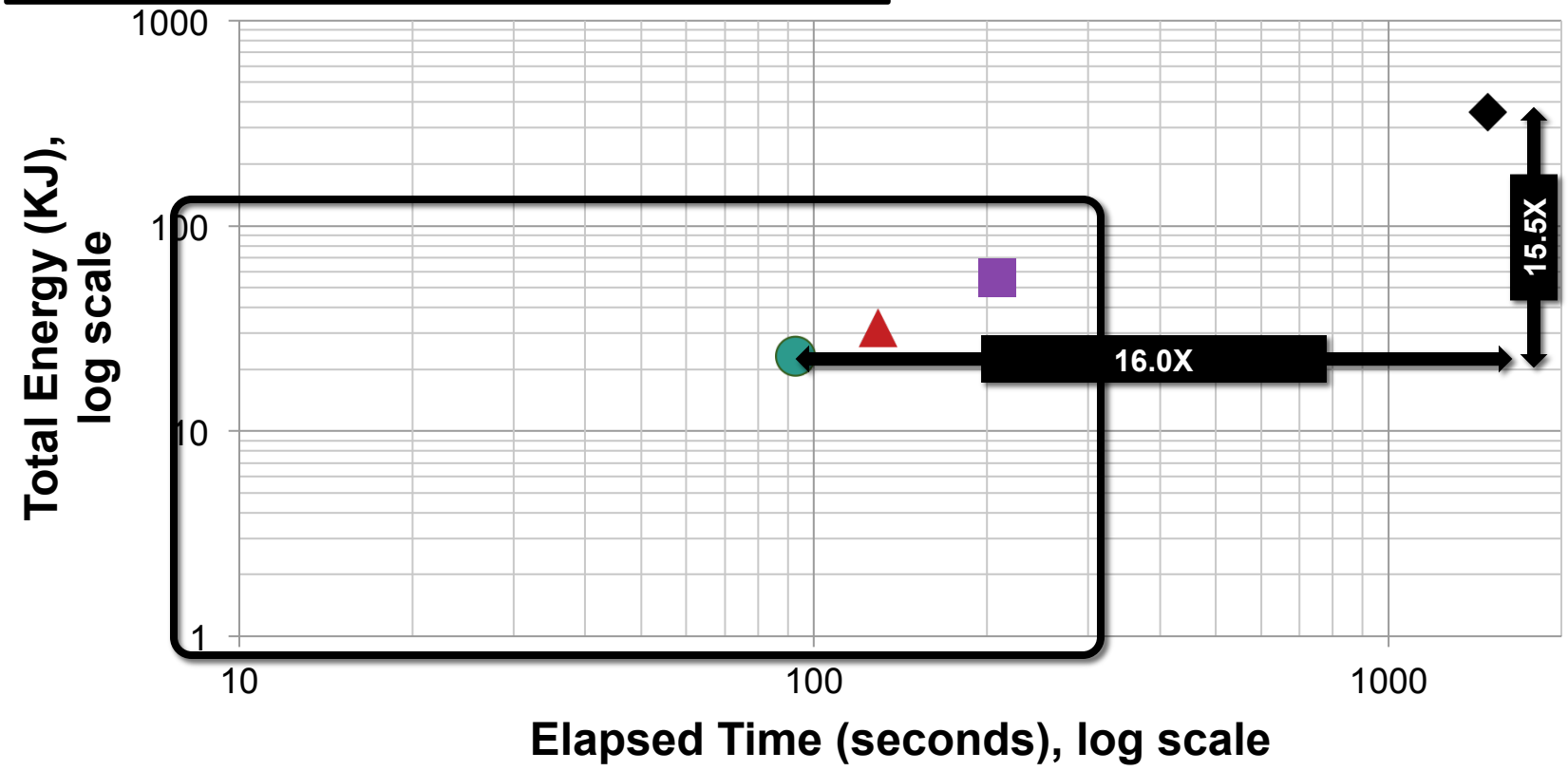
Experimental Setup: TPC-H

Modified TPC-H Lineitem Table, 600M tuples, 94GB

```
Lineitem (
    L_ORDERKEY          bigint,
    L_PARTKEY           int,
    L_SUPPKEY           int,
    L_LINENUMBER        int,
    L_QUANTITY          bigint,
    L_EXTENDEDPRICE     bigint,
    L_DISCOUNT          bigint,
    L_TAX               bigint,
    L_SHIPDATE          int,
    L_COMMITDATE        int,
    L_RECEIPTDATE       int,
    L_RETURNFLAG        char(1),
    L_LINESTATUS        char(1),
    L_SHIPINSTRUCT      char(25),
    L_SHIPMODE          char(10),
    L_COMMENT           char(47),
)
```

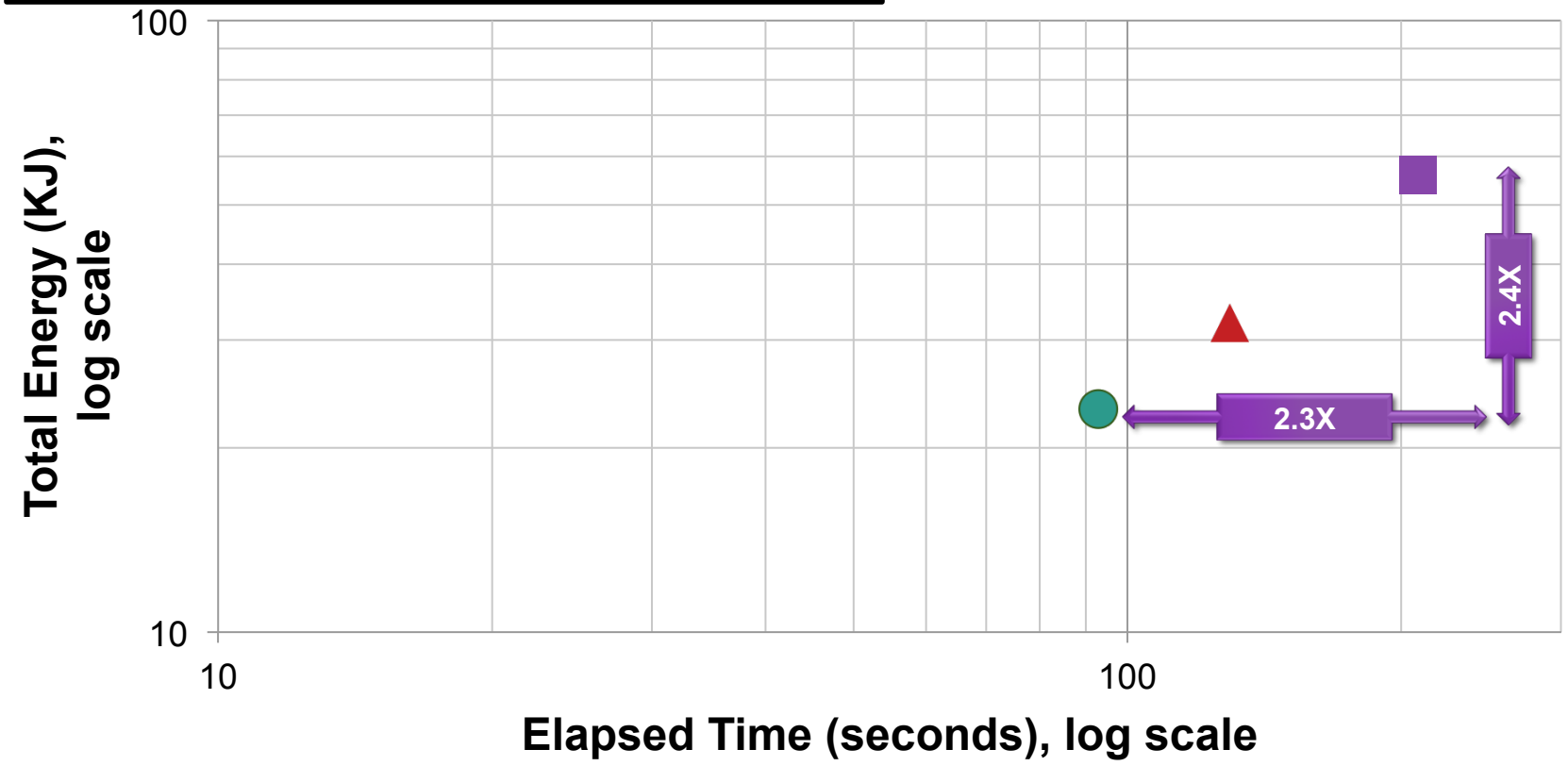
10% Selection Query

```
SELECT    COLUMN_2
FROM      Synthetic64
WHERE     COLUMN_1 < [VALUE]
```



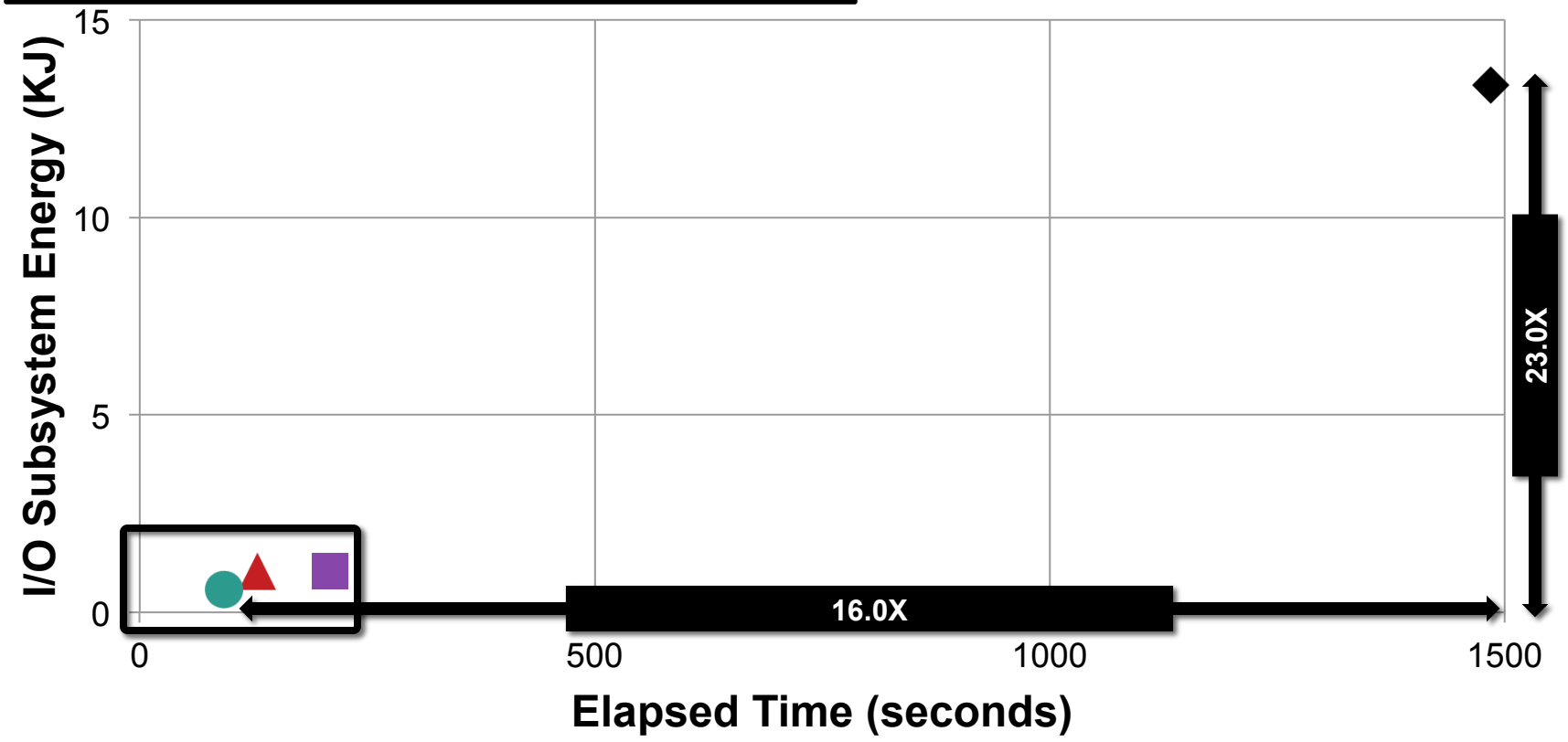
10% Selection Query

```
SELECT    COLUMN_2
FROM      Synthetic64
WHERE     COLUMN_1 < [VALUE]
```



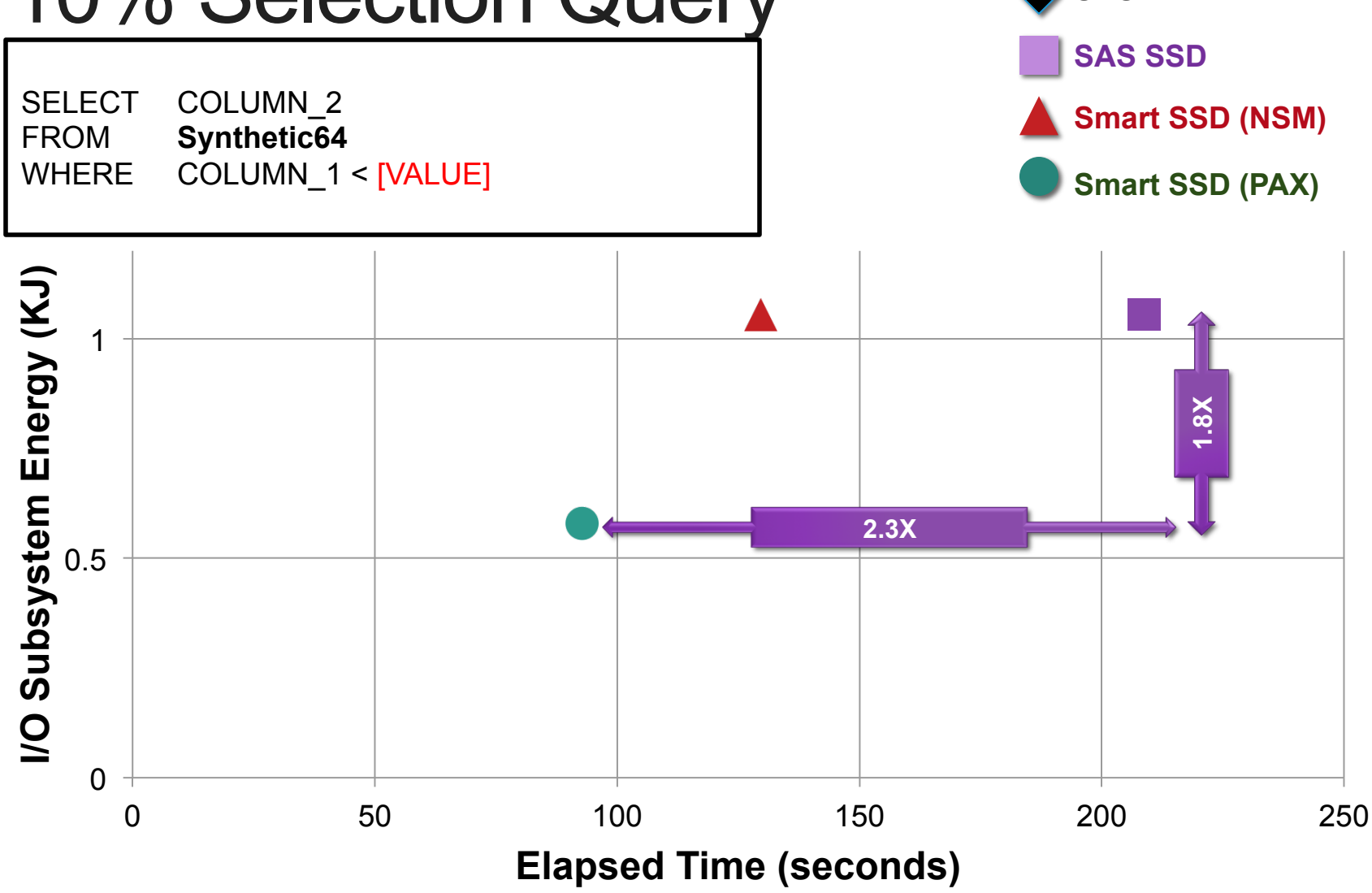
10% Selection Query

```
SELECT    COLUMN_2
FROM      Synthetic64
WHERE     COLUMN_1 < [VALUE]
```



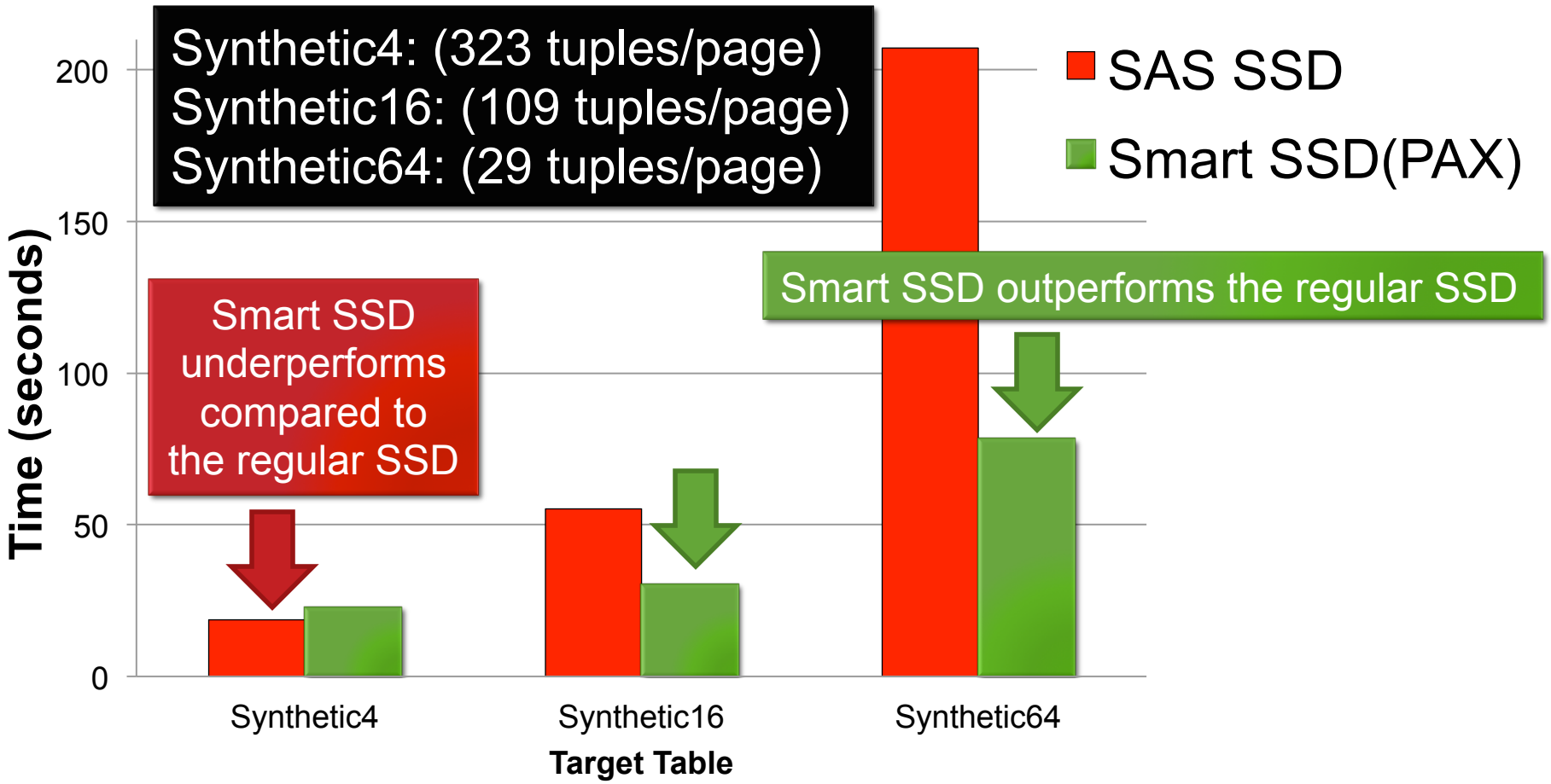
10% Selection Query

```
SELECT    COLUMN_2
FROM      Synthetic64
WHERE     COLUMN_1 < [VALUE]
```



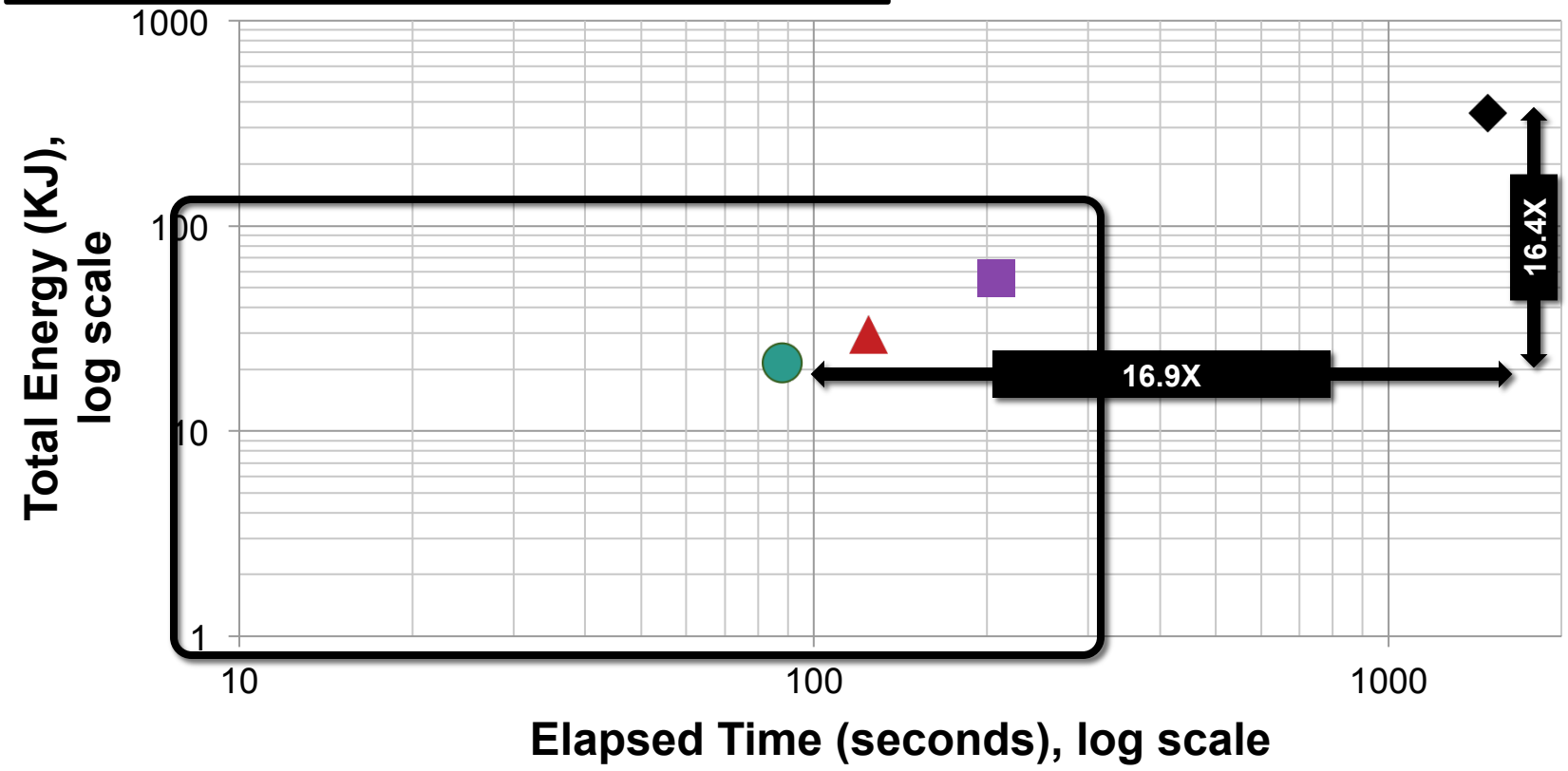
Effect of the # of Tuples on a Page

SELECT COLUMN_2, FROM [Synthetic_table] WHERE COLUMN_1 < [VALUE]



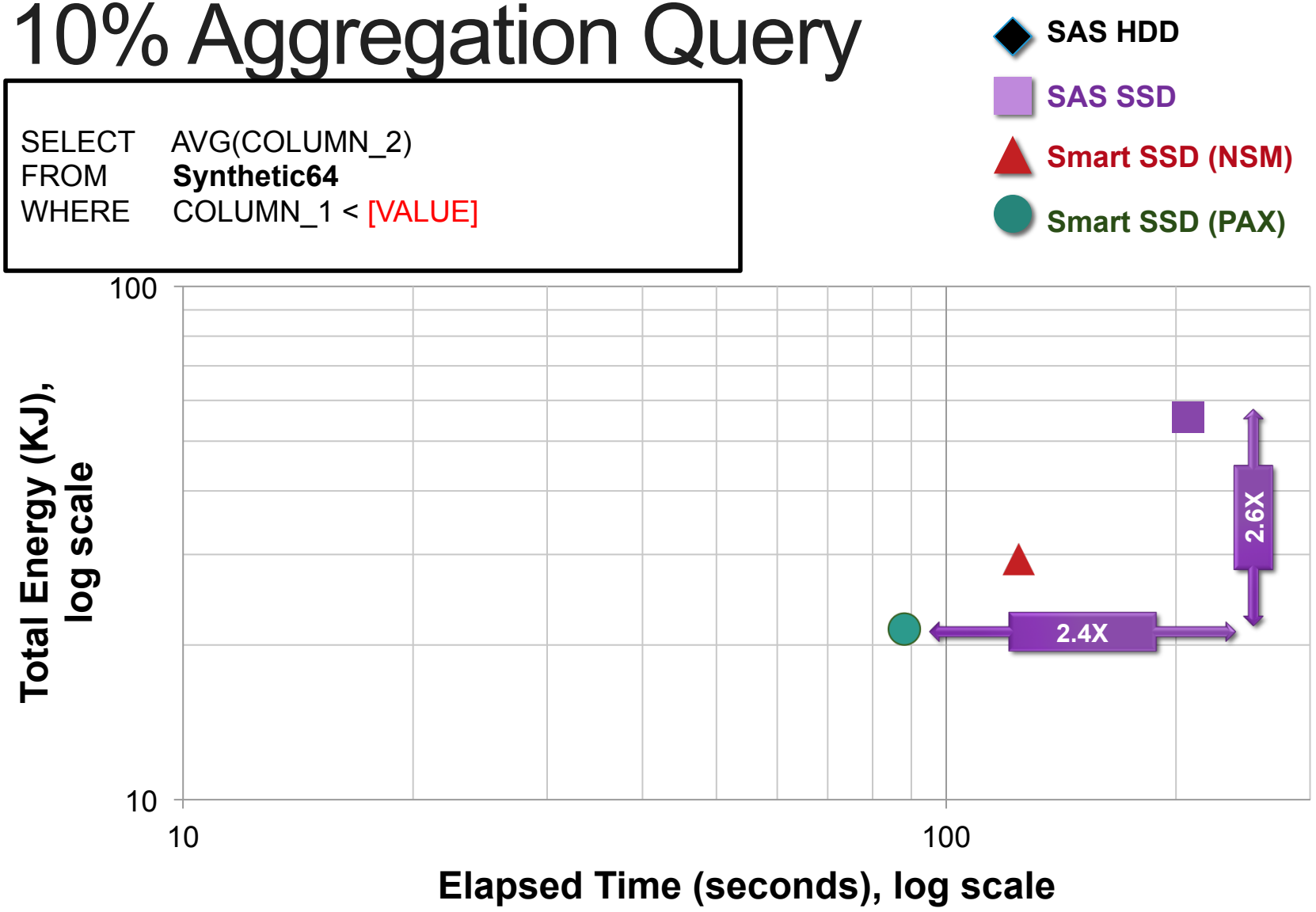
10% Aggregation Query

```
SELECT    AVG(COLUMN_2)
FROM      Synthetic64
WHERE     COLUMN_1 < [VALUE]
```



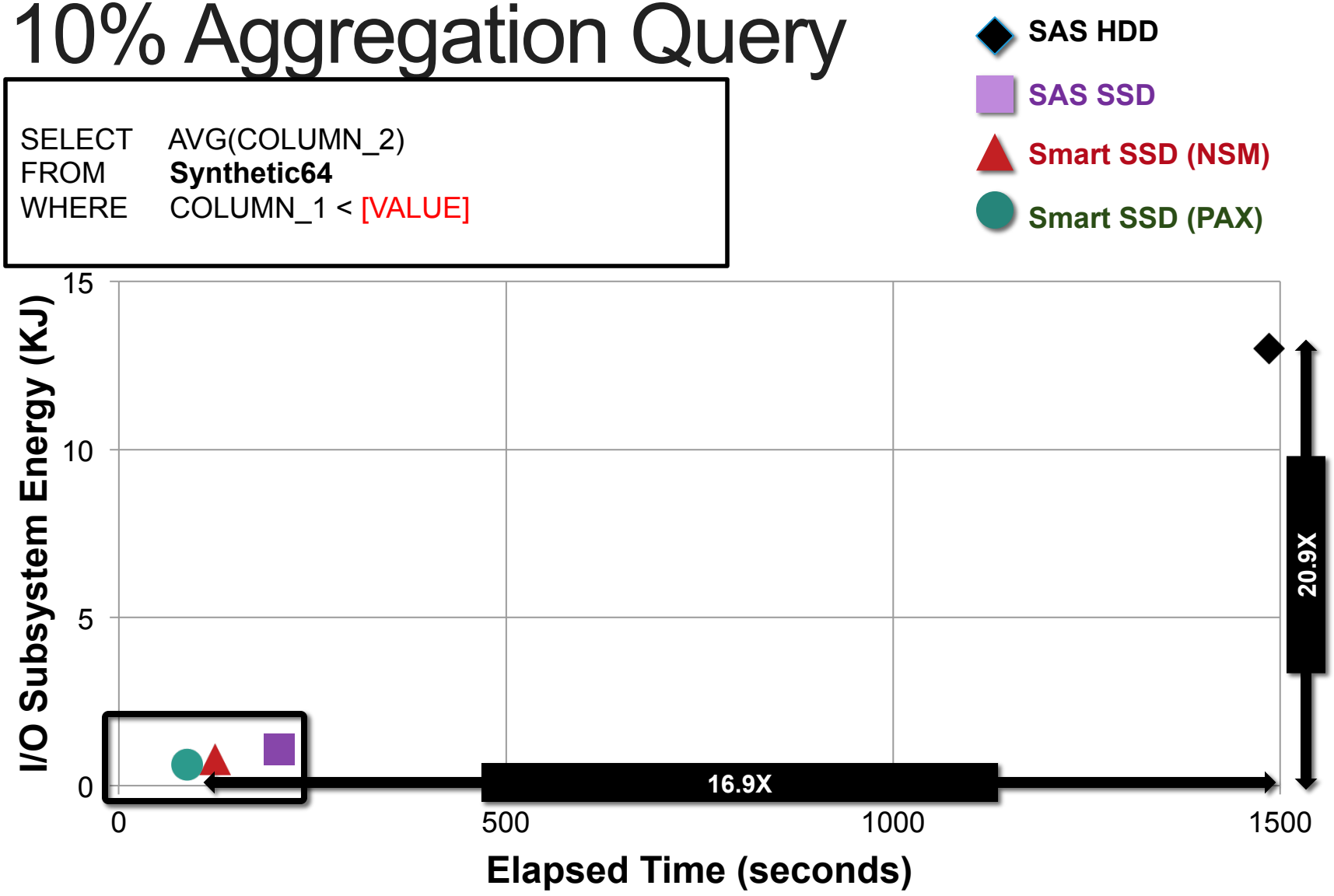
10% Aggregation Query

```
SELECT    AVG(COLUMN_2)
FROM      Synthetic64
WHERE     COLUMN_1 < [VALUE]
```



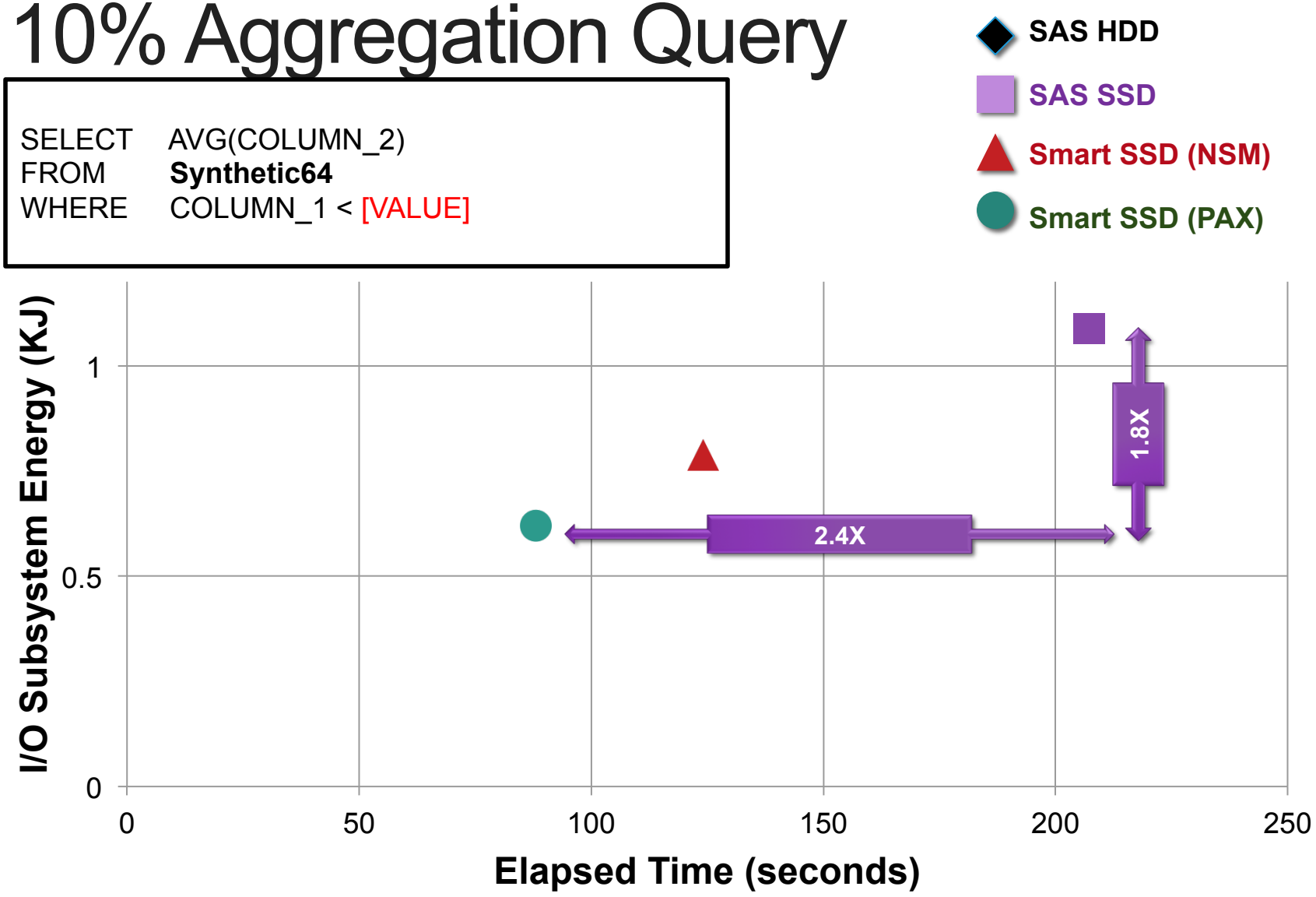
10% Aggregation Query

```
SELECT    AVG(COLUMN_2)
FROM      Synthetic64
WHERE     COLUMN_1 < [VALUE]
```



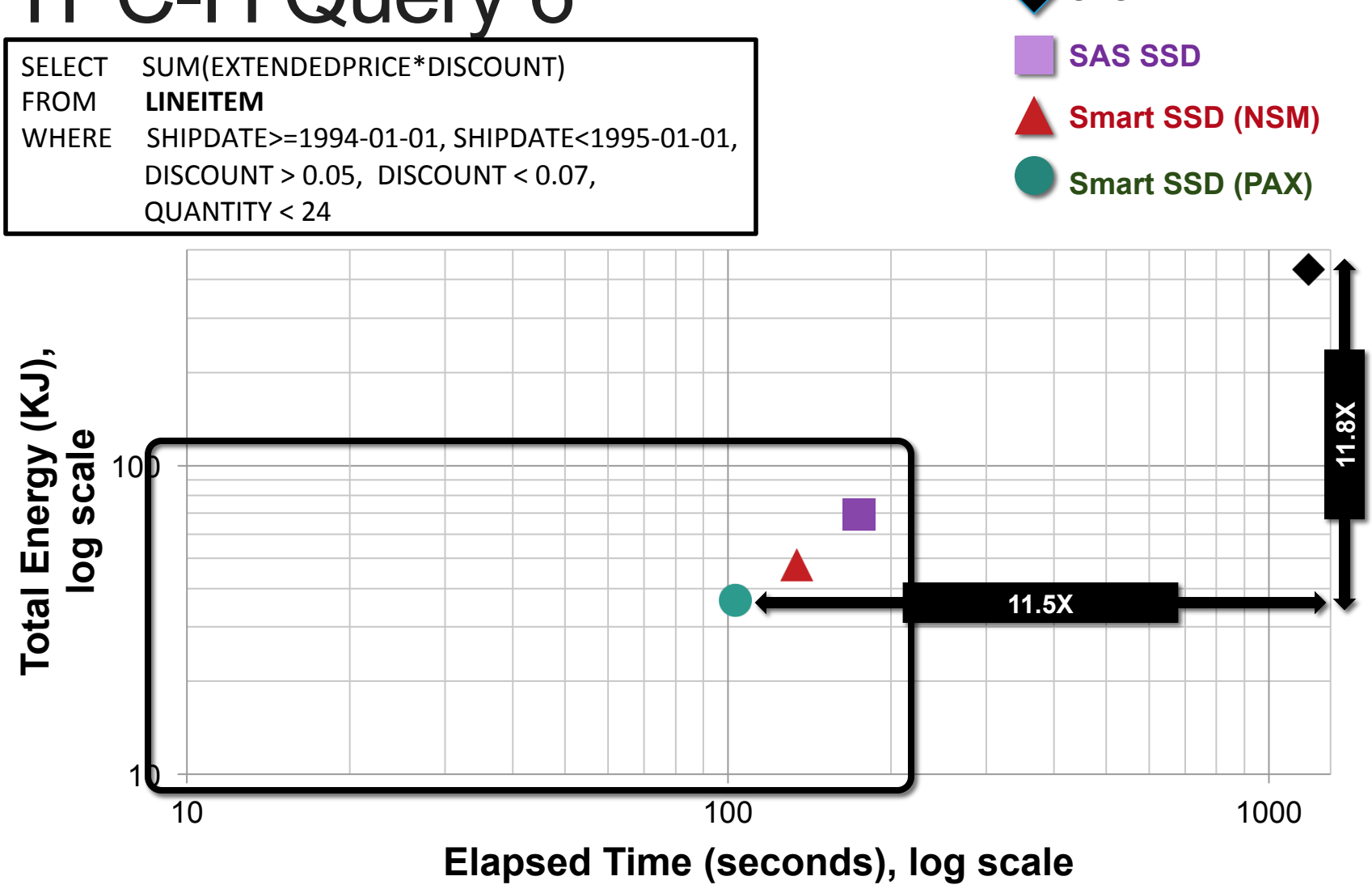
10% Aggregation Query

```
SELECT    AVG(COLUMN_2)
FROM      Synthetic64
WHERE     COLUMN_1 < [VALUE]
```



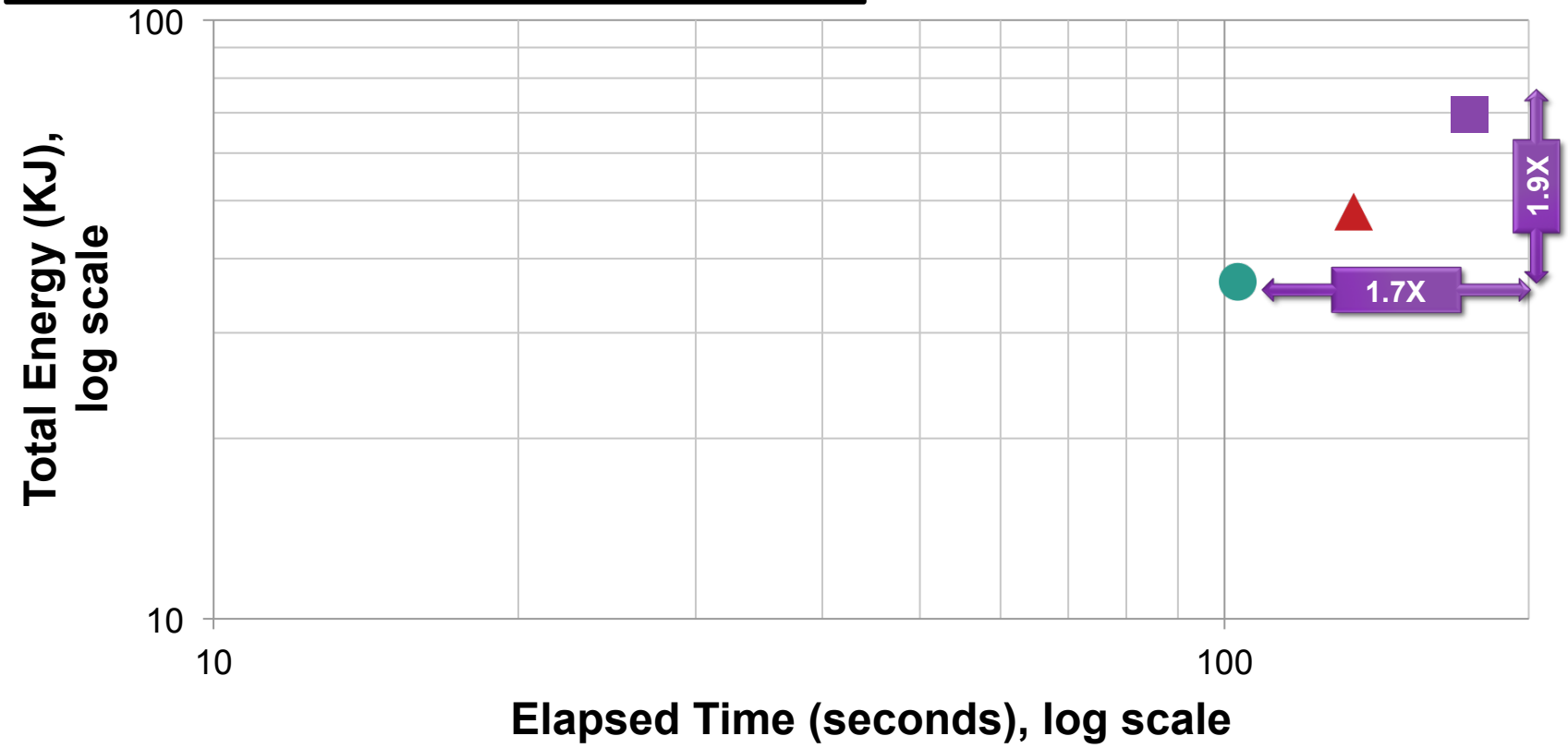
TPC-H Query 6

```
SELECT    SUM(EXTENDEDPRICE*DISCOUNT)
FROM      LINEITEM
WHERE     SHIPDATE>=1994-01-01, SHIPDATE<1995-01-01,
          DISCOUNT > 0.05,  DISCOUNT < 0.07,
          QUANTITY < 24
```



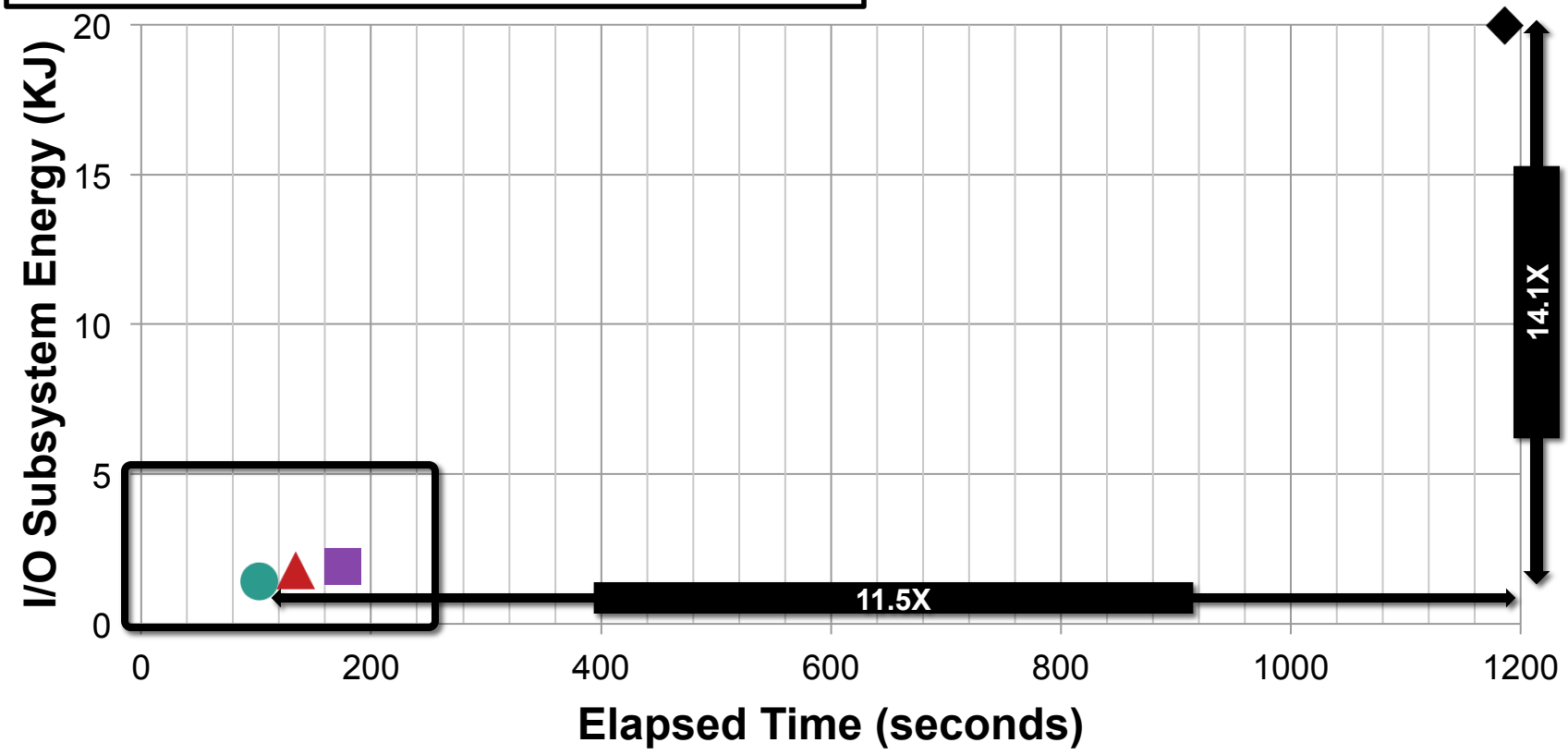
TPC-H Query 6

```
SELECT    SUM(EXTENDEDPRICE*DISCOUNT)
FROM      LINEITEM
WHERE     SHIPDATE>=1994-01-01, SHIPDATE<1995-01-01,
          DISCOUNT > 0.05,  DISCOUNT < 0.07,
          QUANTITY < 24
```



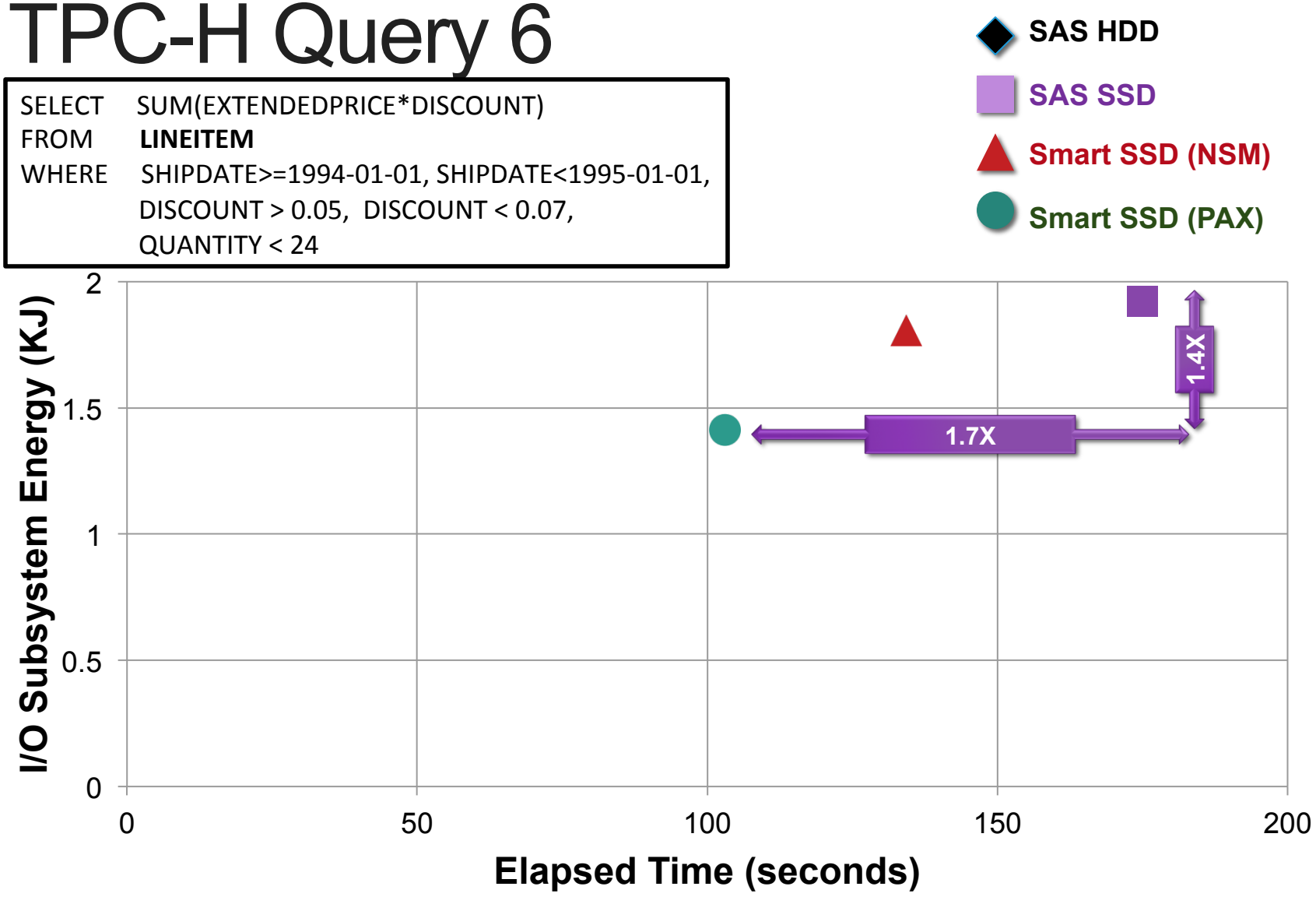
TPC-H Query 6

```
SELECT    SUM(EXTENDEDPRICE*DISCOUNT)
FROM      LINEITEM
WHERE     SHIPDATE>=1994-01-01, SHIPDATE<1995-01-01,
          DISCOUNT > 0.05,  DISCOUNT < 0.07,
          QUANTITY < 24
```



TPC-H Query 6

```
SELECT    SUM(EXTENDEDPRICE*DISCOUNT)
FROM      LINEITEM
WHERE     SHIPDATE>=1994-01-01, SHIPDATE<1995-01-01,
          DISCOUNT > 0.05,  DISCOUNT < 0.07,
          QUANTITY < 24
```



Conclusions and Directions for Future Work

Big Picture

- Computing and storage boundaries are no longer as rigid as they have been in the past
- "Commodity" computing in storage devices is here
- Communication buses evolve slower than other parts of the system

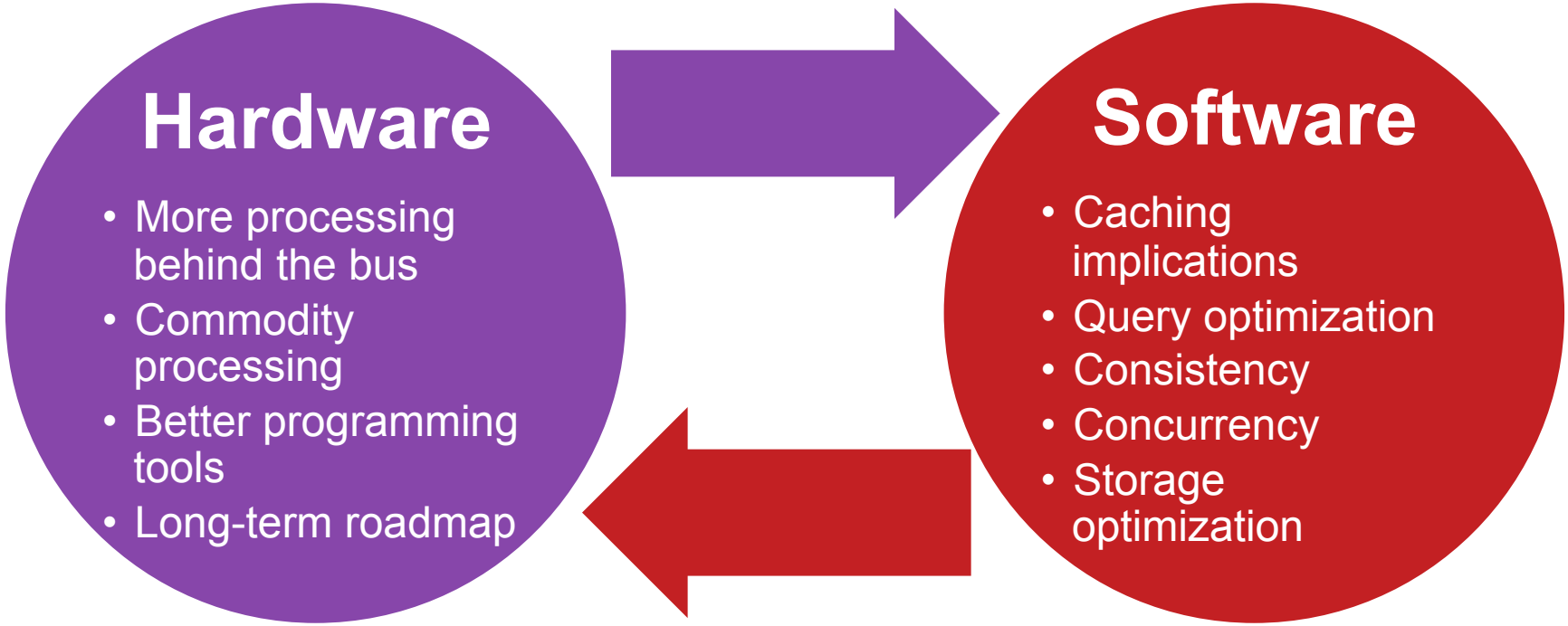
Opportunity

- Push code to data rather than pull data to the (main) processor
- Make general-purpose processing a commodity pre-built "into" the device

Challenges

- Need better programming tools
- Various SSD hardware architectural changes needed (but appear to be feasible, cheaply)
- DBMS internals also need significant changes

Co-dependency



- **Significant $ and time investment needed on both sides.**
- **Who is willing to take the long-term view and invest?**