TD 1: Performance Metrics and Size-Up

Part 1 – Theoretical Exercises

Exercise 1 – Size-Up Analysis

Formula:

$T\_ideal(n; p) = q(n) / (p \times V)$

We want $T\_ideal(n; p) = T\_ideal(n_0; 1)$:

$q(n) / (p \times V) = q(n_0) / V \rightarrow p = q(n) / q(n_0)$

Example: If $q(n) \propto n^2$:

$p = (n / n_0)^2$

For $n = 2n_0 \rightarrow p = 4$

For $n = 3n_0 \rightarrow p = 9$

Exercise 2 – Comparing Workload Growth

For each workload:

$q_1(n) \propto n \rightarrow$ doubling $n \rightarrow p = 2$

$q_2(n) \propto n^2 \rightarrow$ doubling $n \rightarrow p = 4$

$q_3(n) \propto n^3 \rightarrow$ doubling $n \rightarrow p = 8$

Exercise 3 – Practical Limits

Ideal curve: linear : speedup proportional to p.

Realistic curve: below ideal due to communication and synchronization overhead.

Poor scaling: curve flattens as p increases because overhead dominates.

Part 2 – Applied Big Data Exercises

Exercise A – Storage & Processing Needs

1) Total events per day: $200{,}000 \times 86{,}400 = 17.28 \times 10^9$ events

   Total data: $17.28 \times 10^9 \times 500$ bytes $= 8.64 \times 10^{12}$ bytes ≈ 8.64 TB

2) Processing time on 1 server: 8.64 TB / 100 MB/s

   Convert: 8.64 TB $= 8.64 \times 10^6$ MB $\rightarrow 8.64 \times 10^6 / 100 = 86{,}400$ s ≈ 24 hours

3) Processing time on 20 servers: 24 h / 20 = 1.2 h

4) A single server is impractical due to long processing times.

Exercise B – Parallel Processing & Speedup

Dataset: 10 TB $= 10 \times 10^6$ MB

Node speed: 200 MB/s

1) Time with 5 nodes: $10 \times 10^6 / (5 \times 200) = 10{,}000$ s ≈ 2.78 h

2) Time with 20 nodes (ideal): $10 \times 10^6 / (20 \times 200) = 2{,}500$ s ≈ 0.69 h

3) Speedup = T5 / T20 = 2.78 / 0.69 ≈ 4
Efficiency = Speedup / (20 / 5) = 4 / 4 = 1 (ideal)
With 10% overhead: Effective speedup = 4 × 0.9 = 3.6 → Efficiency = 3.6 / 4 = 0.9 (90%)

Show that overhead reduces real efficiency.

Exercise C – Scalability Challenge

Original workload: 5 TB with 10 nodes.
New workload: 25 TB (5× larger).

1) Ideal nodes = 10 × 5 = 50 nodes
2) With 20% overhead: Effective capacity = 50 × (1 - 0.20) = 40 nodes → Need more to compensate.
Required nodes ≈ 50 / 0.8 = 62.5 → about 63 nodes

3) Scaling is harder because overhead grows with cluster size.