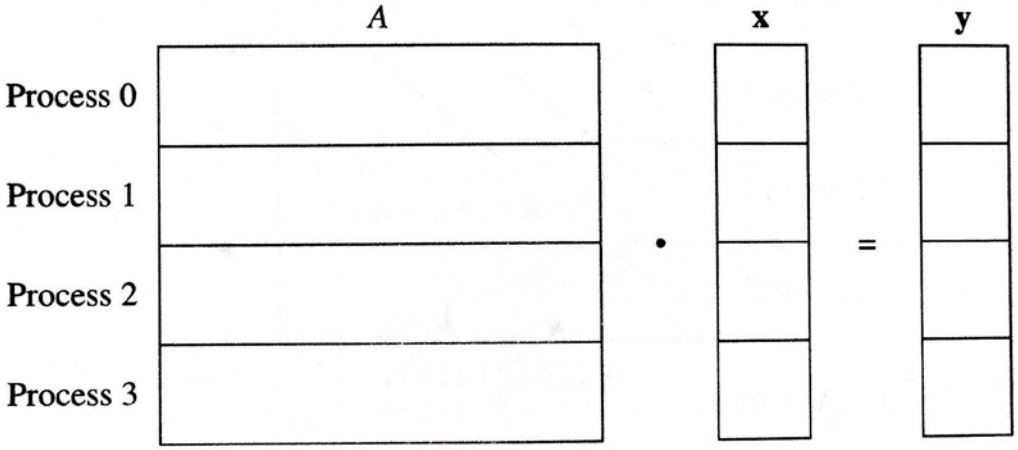
# Matrix Vector Mulitplication

Professor Seokkoo Kang
Department of Civil & Environmental Engineering
kangsk78@hanyang.ac.kr

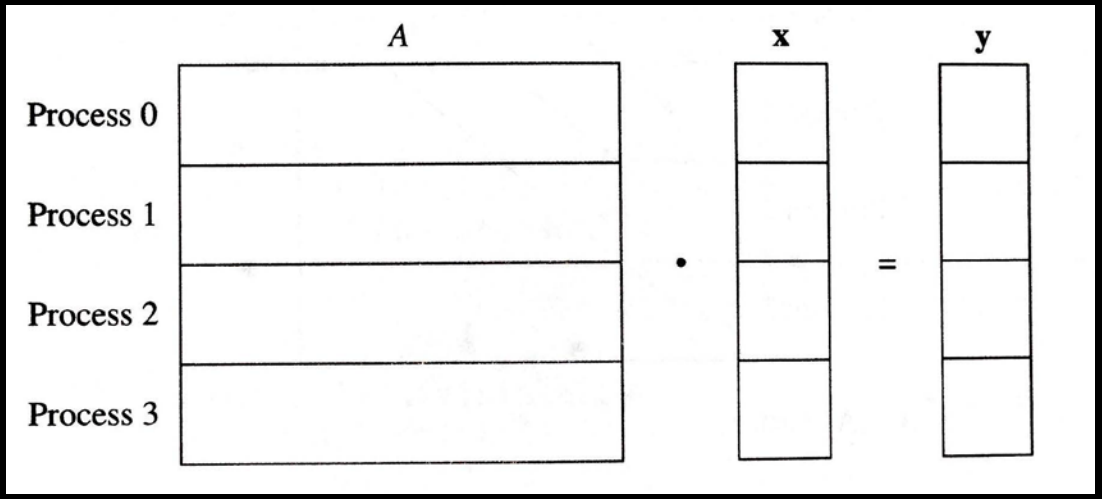
# Block-row distribution

| *Process* | *Elements of A* | | | |
|---|---|---|---|---|
| 0 | $a_{00}$ | $a_{01}$ | $a_{02}$ | $a_{03}$ |
| | $a_{10}$ | $a_{11}$ | $a_{12}$ | $a_{13}$ |
| 1 | $a_{20}$ | $a_{21}$ | $a_{22}$ | $a_{23}$ |
| | $a_{30}$ | $a_{31}$ | $a_{32}$ | $a_{33}$ |
| 2 | $a_{40}$ | $a_{41}$ | $a_{42}$ | $a_{43}$ |
| | $a_{50}$ | $a_{51}$ | $a_{52}$ | $a_{53}$ |
| 3 | $a_{60}$ | $a_{61}$ | $a_{62}$ | $a_{63}$ |
| | $a_{70}$ | $a_{71}$ | $a_{72}$ | $a_{73}$ |

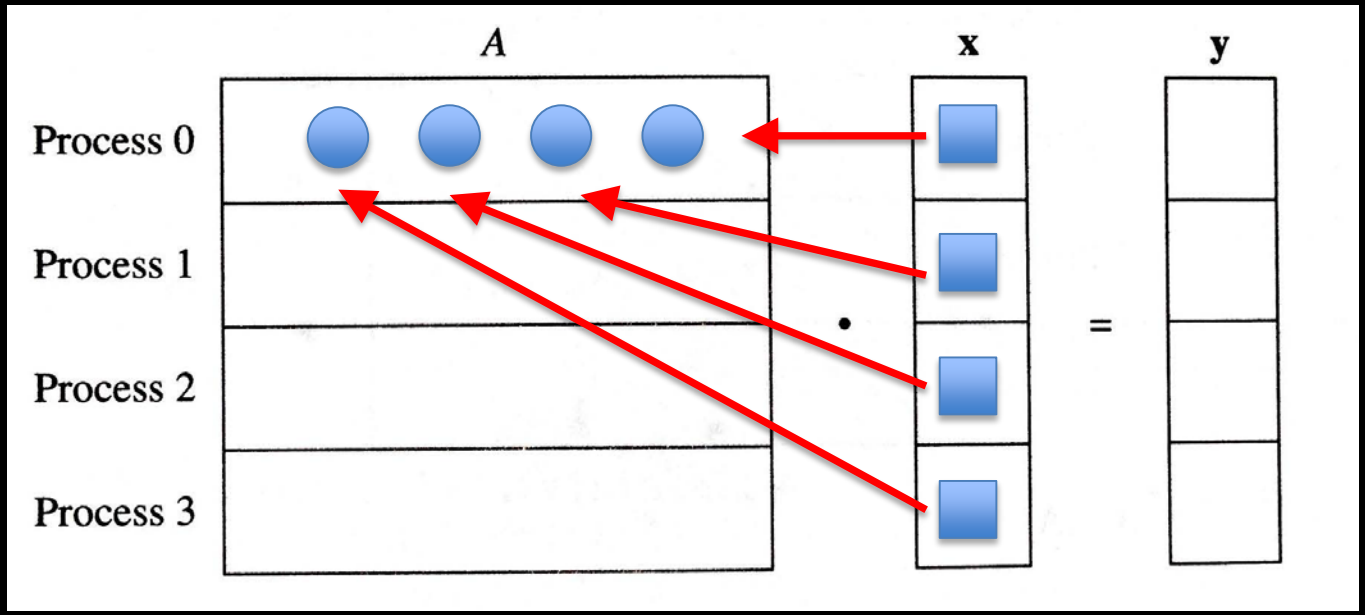


# Methods for calculating y=Ax

- Method 1
  - Gather all of x onto each process
- Method 2
  - Scatter each row of A across the process

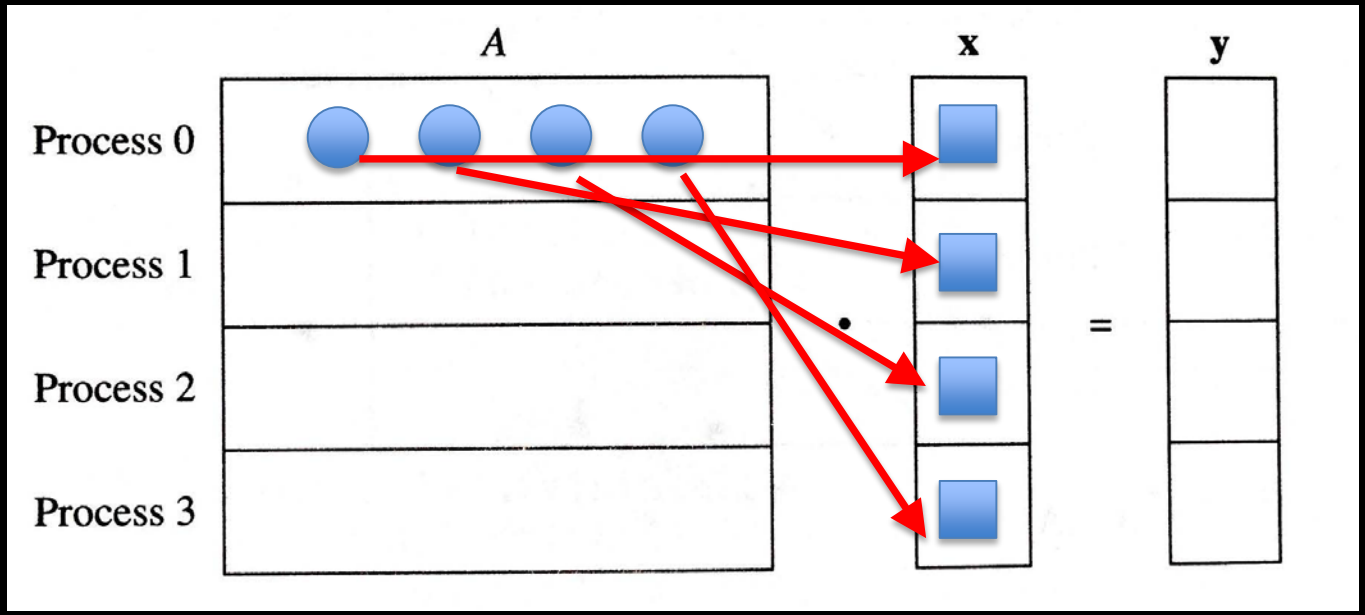


# Method 1




Using MPI_Gather(), gather the entire vector onto my rank. Calculate the local dot products and store the result in y.

# Method 2




Using MPI_Scatter(), scatter the rows across the processors. Calculate the local dot products and store the result in y.

# Assignment

- Write a parallel that calculates a matrix vector product for an arbitrary matrix size of N*N and vector size N.
- Fill the matrix and vector with random numbers.
- You may use either the method 1 or 2.
- Verify the result for small N and test the result for different np and check the result is consistent.
- Measure and report the execution time for N=10,000 matrix for np=1,2,3,4,..,8
- Use MPI_Scatterv and MPI_Gatherv