SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

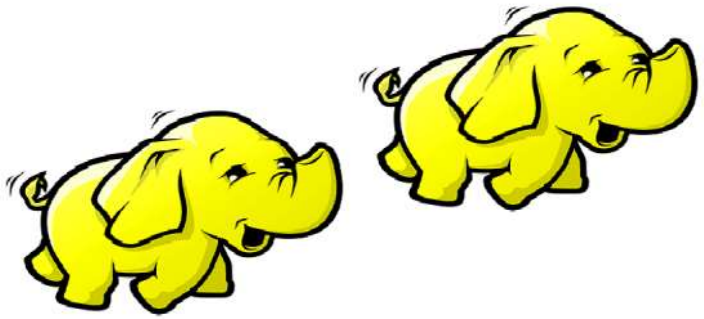
DEPARTMENT OF MCA

I YEAR II SEM

23CAE717 – Cloud Computing

UNIT IV – PROGRAMMING MODEL

Topic 24: Mapping Application

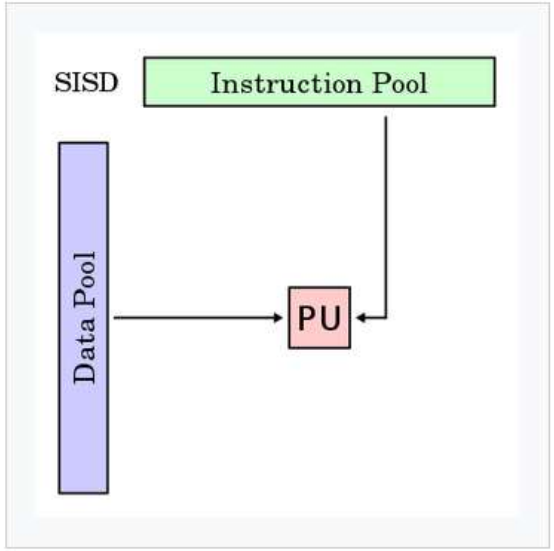


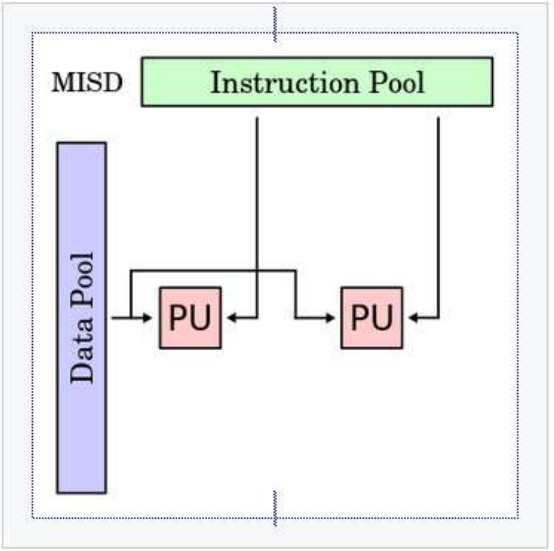
Parallel System Architectures

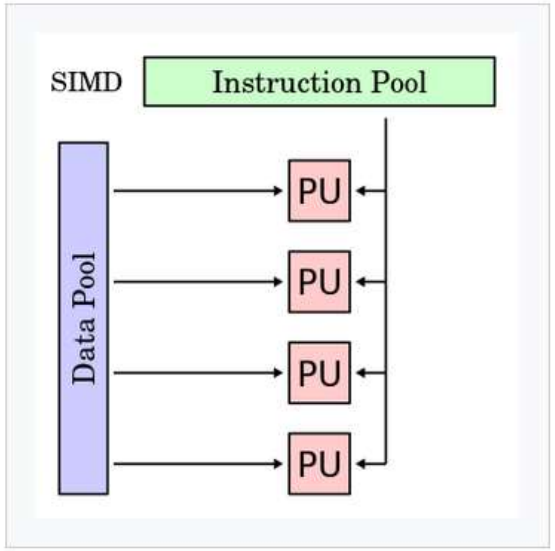
- Jobs are broken into discrete parts that can be executed concurrently
- Parallel systems are programmed in such a way that all CPUs to be coordinated synchronized
- Parallel systems deal with the simultaneous use of
  - multiple computer resources that can include a single computer with multiple processors,
  - a number of computers connected by a network to form a parallel processing cluster or a combination of boot

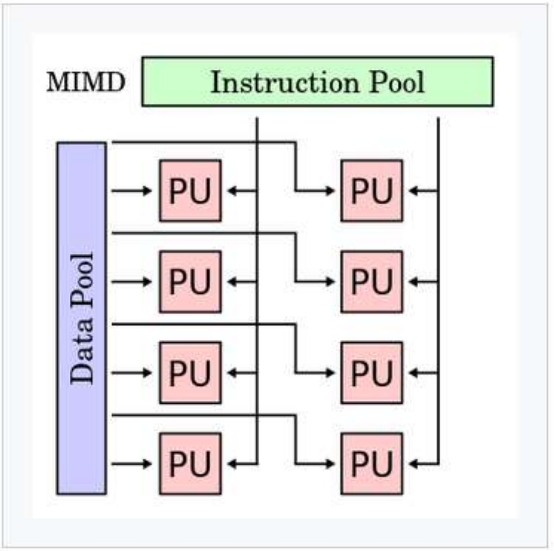
Parallel System Architectures

Data Streams \ Instruction Streams	one	many
one	**SISD** traditional von Neumann single CPU computer	**MISD** May be pipelined Computers
many	**SIMD** Vector processors fine grained data Parallel computers	**MIMD** Multi computers Multiprocessors









SIMD (Single Instruction, Multiple Data)

- All parallel units of computing share the same instruction, but they carry it out on different data elements
- widely used for **3D graphics and audio/video processing** in multimedia applications

MIMD (Multiple Instruction, Multiple Data)

- All parallel units have separate instructions, so each of them can do something different at any given time; one may be adding, another multiplying, yet another evaluating a branch condition, and so on

Mapping Applications

- Five categories of application architectures

Class	Description	Machine Architecture
Synchronous	❑ The problem class can be implemented with Instruction - level lockstep operation as in SIMD architectures	SIMD

Mapping Applications

- Five categories of application architectures

Class	Description	Machine Architecture
Loosely synchronous	❑ It exhibit iterative compute – communication stages with independent compute (map) operations for each CPU that are synchronized with a communication step ❑ Covers many successful MPI applications including partial differential equation solutions and particle dynamics applications.	MIMD on MPP (massively parallel processor)

Mapping Applications

- Five categories of application architectures

Class	Description	Machine Architecture
Asynchronous	❑ Illustrated by Compute Chess and Integer Programming; ❑ Combinatorial search is often supported by dynamic threads. ❑ Important in scientific computing, but it is at the heart of operating systems and concurrency in consumer applications such as MS Word	Shared Memory

Mapping Applications

- Five categories of application architectures

Class	Description	Machine Architecture
Pleasingly parallel	❑ Each component is independent. ❑ In 1988, it estimated this at 20 % of the total number of applications, but that percentage has grown with the use of grids and data analysis applications including	Grids moving to clouds

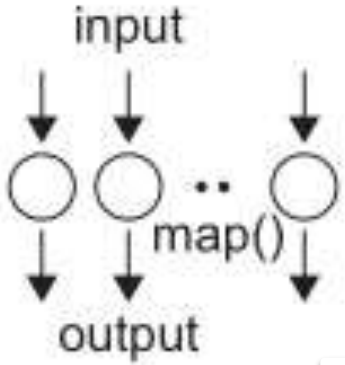
Mapping Applications

Class	Description	Machine Architecture
Meta problems	❑ These are coarse-grained (asynchronous or data flow) combinations of categories 1-4 and 6. ❑ This area has also grown in importance and is well supported by grids and described by workflow	Grids of Clusters

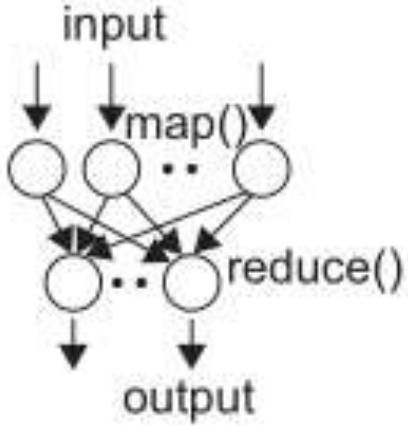
Mapping Applications

Class	Description	Machine Architecture
MapReduce++ (Twister)	❑ This describes file (database) to file (database) operations which have three subcategories ❑ a) Pleasingly Parallel Map Only (similar to 4) ❑ b) Map followed by reductions ❑ c) Iterative "Map followed by reductions"	Data-intensive clouds a) Master- Worker / MapReduce b) MapReduce c) Twister

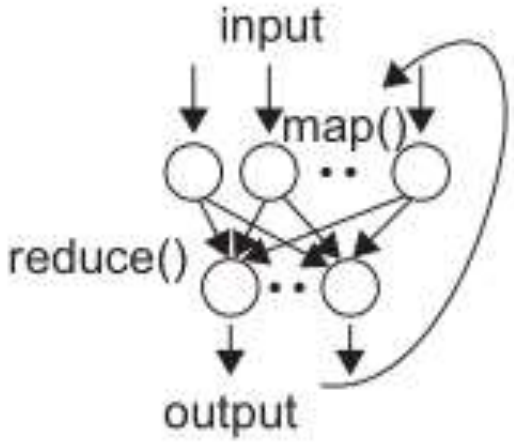
Comparision of MapReduce ++ with loosely synchronous



Map Only



Classic MapReduce



Iterative MapReduce

References

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, “Distributed and Cloud Computing, From Parallel Processing to the Internet of Things”, Morgan Kaufmann Publishers, 2012.

Thank You