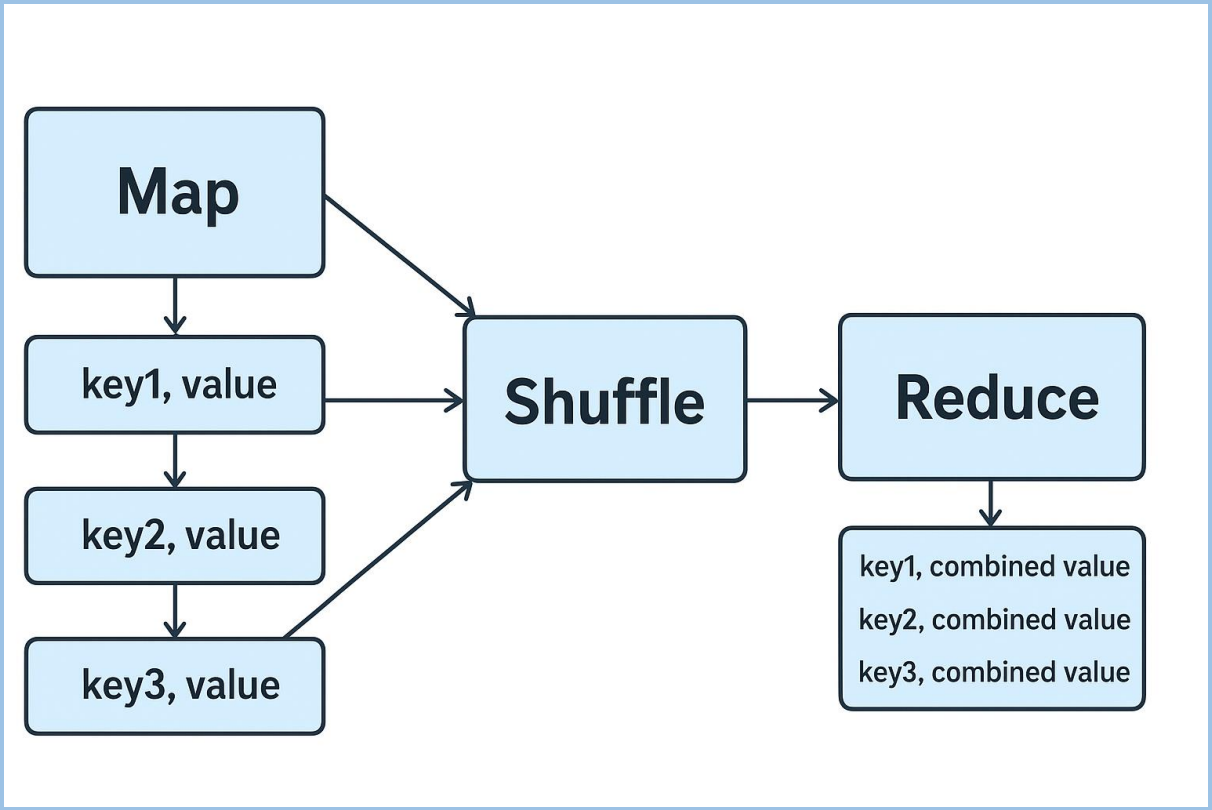
MapReduce in Cloud Computing

Basic Concept

1. MapReduce is a programming model used for processing large data sets.
2. It was originally introduced by Google.
3. It works on the principle of **divide and conquer**.
4. It is mainly used in **cloud computing** and **big data platforms**.
5. It helps to handle **structured, semi-structured, and unstructured data**.

Working Model

6. It divides the work into two stages – **Map** and **Reduce**.
7. **Map stage:** Data is split into chunks and processed in parallel.
8. **Reduce stage:** Results are combined and summarized.
9. Input and output are always represented in the form of **key-value pairs**.
10. The system automatically takes care of data distribution and task scheduling.



Map Phase

11. The map function processes input data.
12. It generates **intermediate key-value pairs**.
13. Example: For word count → (word, 1).
14. Each mapper works on a small block of data.
15. Multiple mappers run at the same time across servers.

Shuffle and Sort

16. After mapping, data is **shuffled** to group similar keys together.
17. Sorting ensures that keys are arranged properly.
18. This step is done automatically by the system.
19. It prepares the data for the reduce stage.
20. Without shuffle and sort, reducers cannot work properly.

Reduce Phase

21. The reducer takes grouped key-value pairs.
22. It combines the values for each key.
23. Example: For word count → (word, [1,1,1,1]) → (word, 4).
24. The reducer outputs the final result.
25. Many reducers can run in parallel for efficiency.

Execution in Cloud

26. Cloud computing provides multiple servers for MapReduce.
27. Data is stored across servers using **distributed file systems** like HDFS.
28. Each server performs map tasks independently.
29. Reduce tasks collect results from multiple servers.
30. The cloud ensures resource allocation and load balancing.

Advantages

31. **Scalability:** Works with terabytes or petabytes of data.
32. **Parallel Processing:** Tasks run simultaneously on many nodes.

33. **Fault Tolerance:** Failed tasks are automatically re-executed.
34. **Flexibility:** Can handle different types of data.
35. **Cost-Effective:** Cloud pay-as-you-go model reduces cost.

Use Cases

36. Word count in large document collections.
37. Analyzing search engine logs.
38. Social media trend analysis.
39. Fraud detection in banking data.
40. Training machine learning models.

Examples

41. Google uses MapReduce for indexing web pages.
42. Yahoo used it in their Hadoop clusters.
43. Facebook applies it for analyzing user activity.
44. Amazon uses it in recommendation engines.
45. Governments use it for census and survey analysis.

Limitations

46. Not suitable for **real-time processing**.
47. Complex algorithms may need multiple MapReduce jobs.
48. Network bottleneck may occur during shuffle phase.
49. Requires good infrastructure for best performance.
50. Alternatives like **Apache Spark** are sometimes preferred for speed.