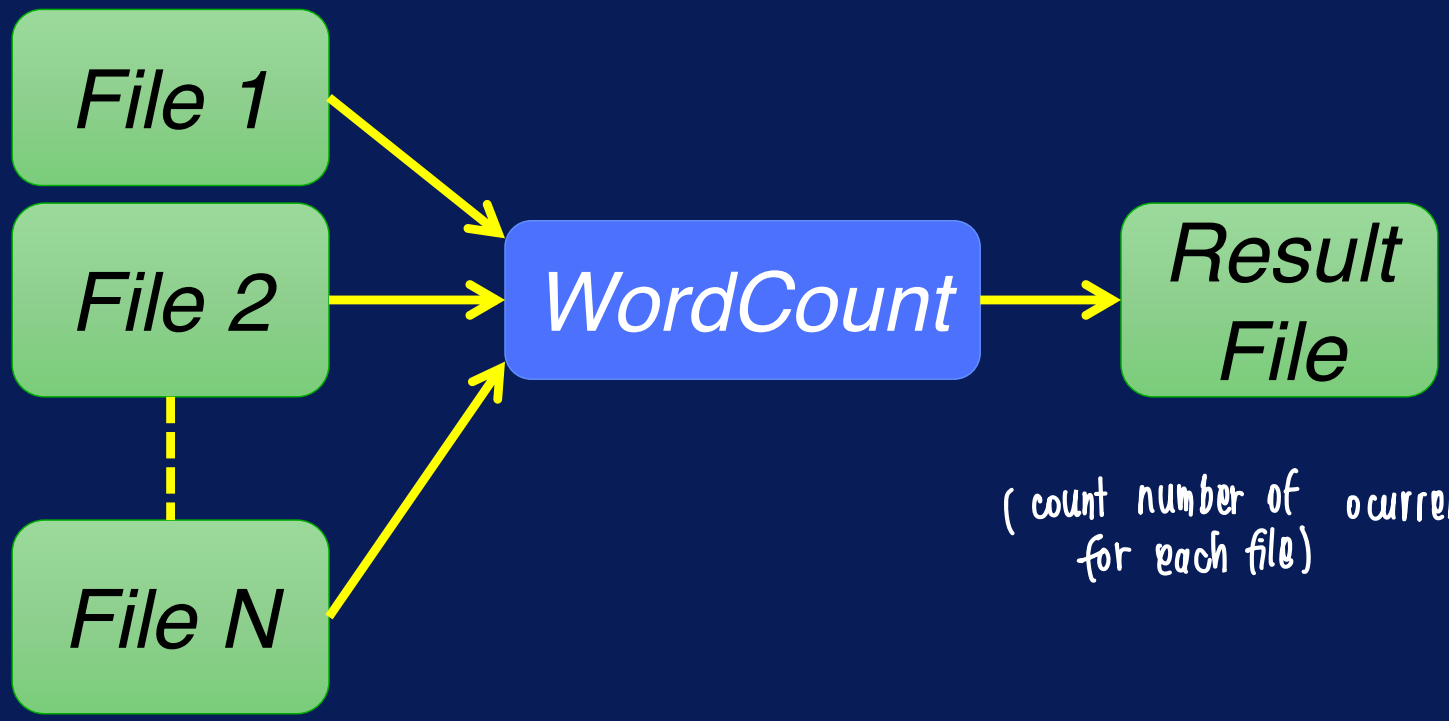
Big Data Processing Pipelines:

A Dataflow Approach

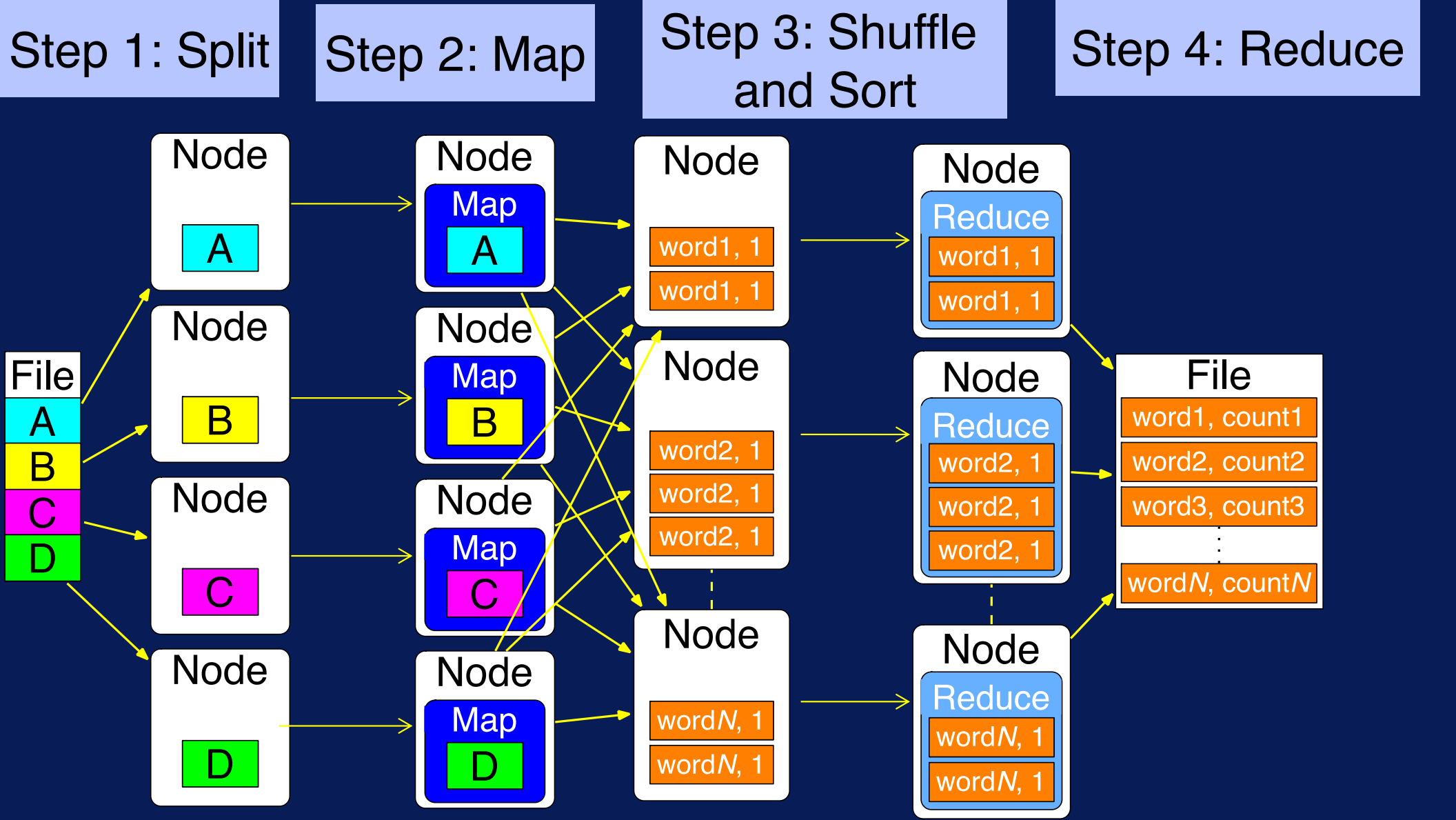
After this video you will be able to..

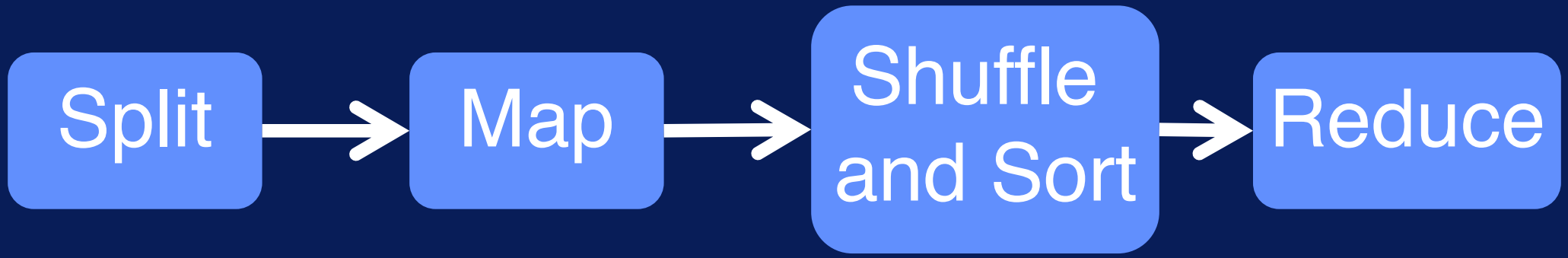
- Summarize what dataflow means and its role in data science
- Explain “split->do->merge” big data pipeline with examples
- Define the terms data parallel

Example MapReduce Application: WordCount



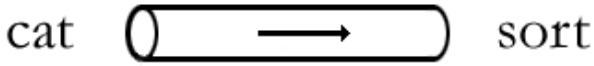
(count number of ocurrences of words for each file)


Step 1: Split
Step 2: Map
Step 3: Shuffle and Sort
Step 4: Reduce
File
A
B
C
D
Node
A
Node
B
Node
C
Node
D
Node
Map
A
Node
Map
B
Node
Map
C
Node
Map
D
Node
word1, 1
word1, 1
Node
word2, 1
word2, 1
word2, 1
Node
wordN, 1
wordN, 1
Node
Reduce
word1, 1
word1, 1
Node
Reduce
word2, 1
word2, 1
word2, 1
Node
Reduce
wordN, 1
wordN, 1
File
word1, count1
word2, count2
word3, count3
wordN, countN


Split
Map
Shuffle and Sort
Reduce

Represents a large number of applications.

Big Data Pipelines

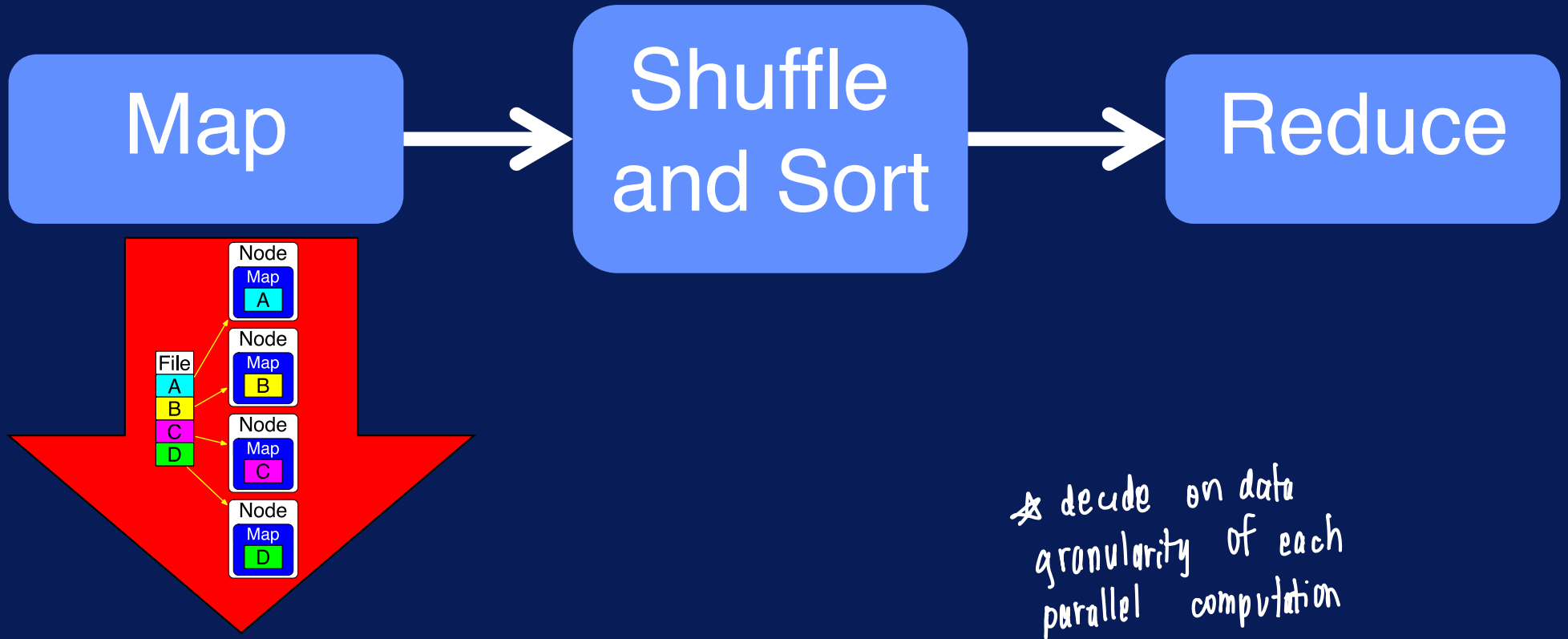


A UNIX pipe provides one-way communication between two processes on the same computer

parallelism: running the same functions simultaneously for the elements or partitions of a dataset on multiple cores

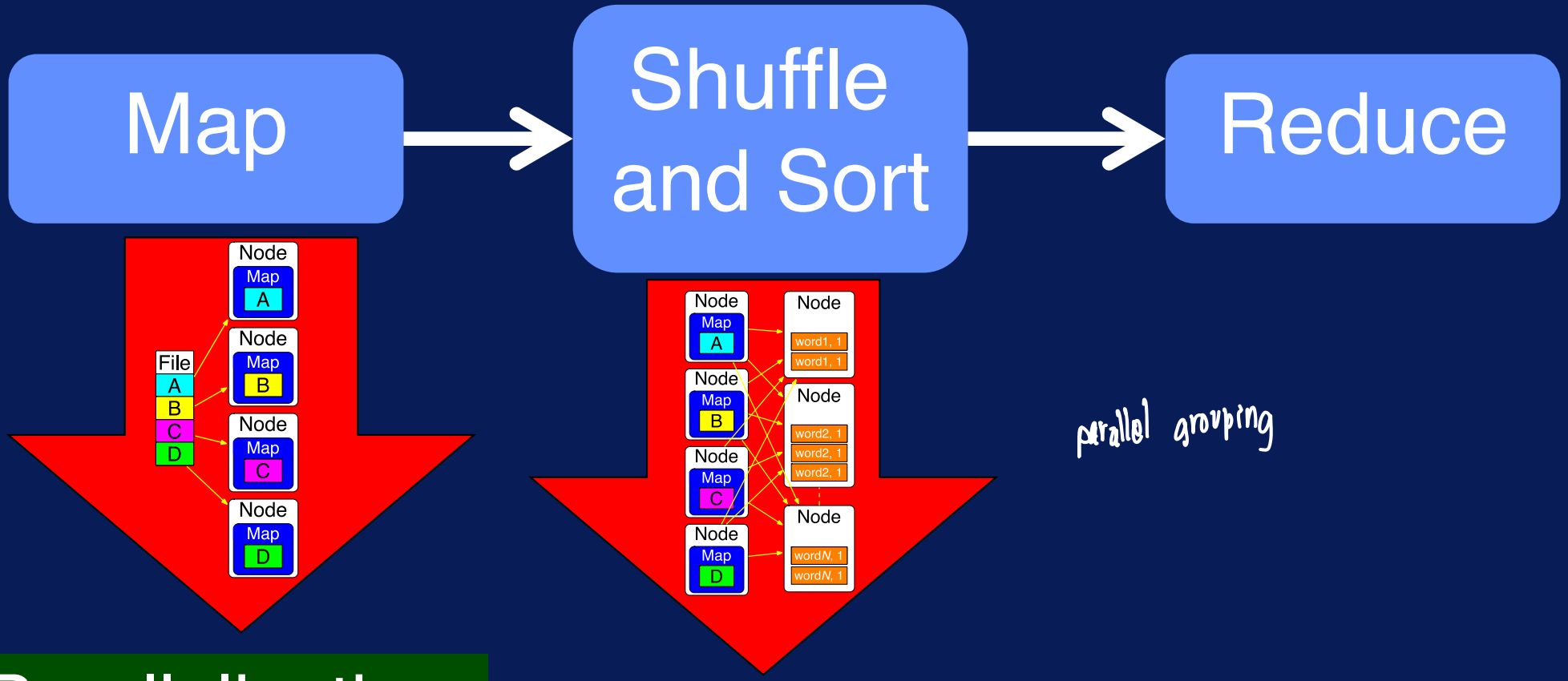
* occurs in every step


Map
Shuffle and Sort
Reduce



Parallelization over the input

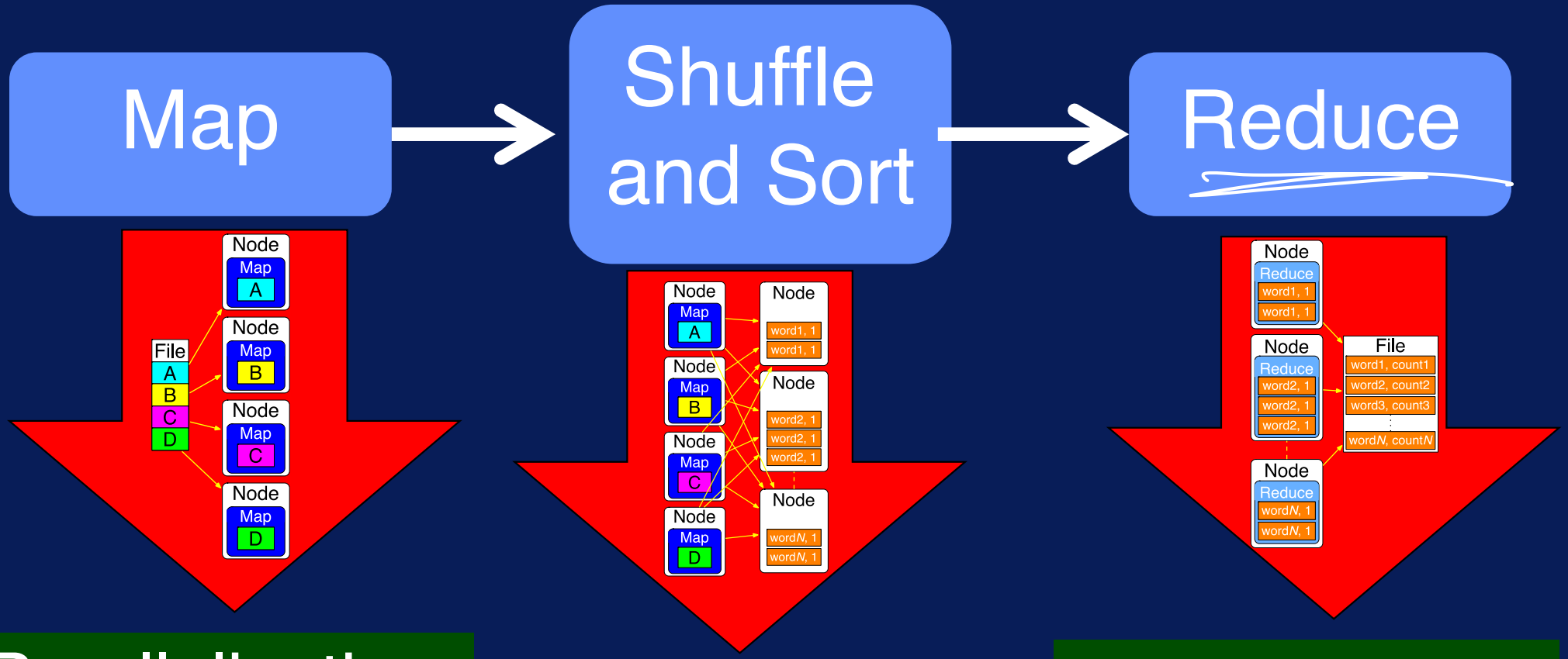
* decide on data granularity of each parallel computation



parallel grouping

Parallelization over the input

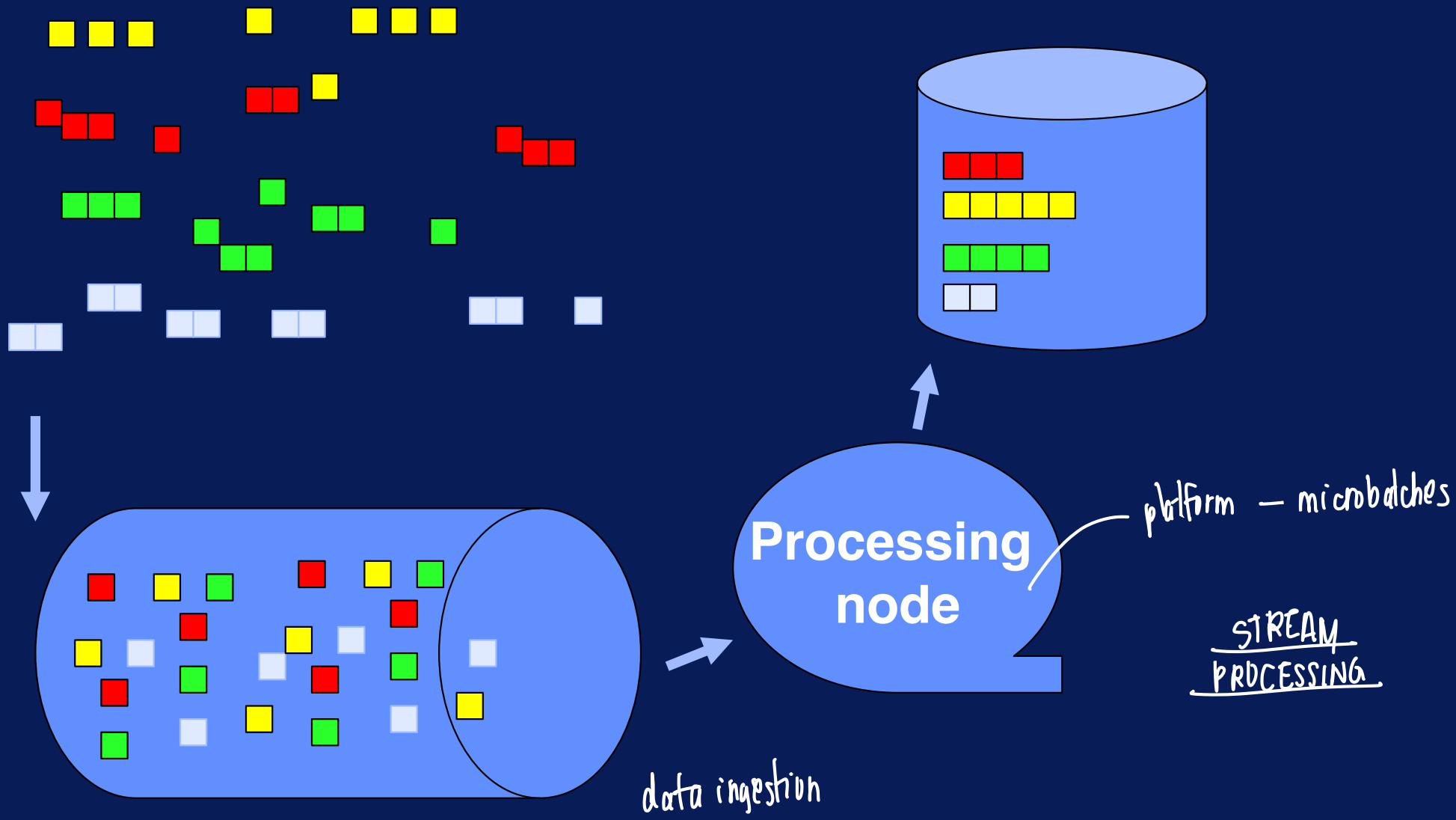
Parallelization data sorting

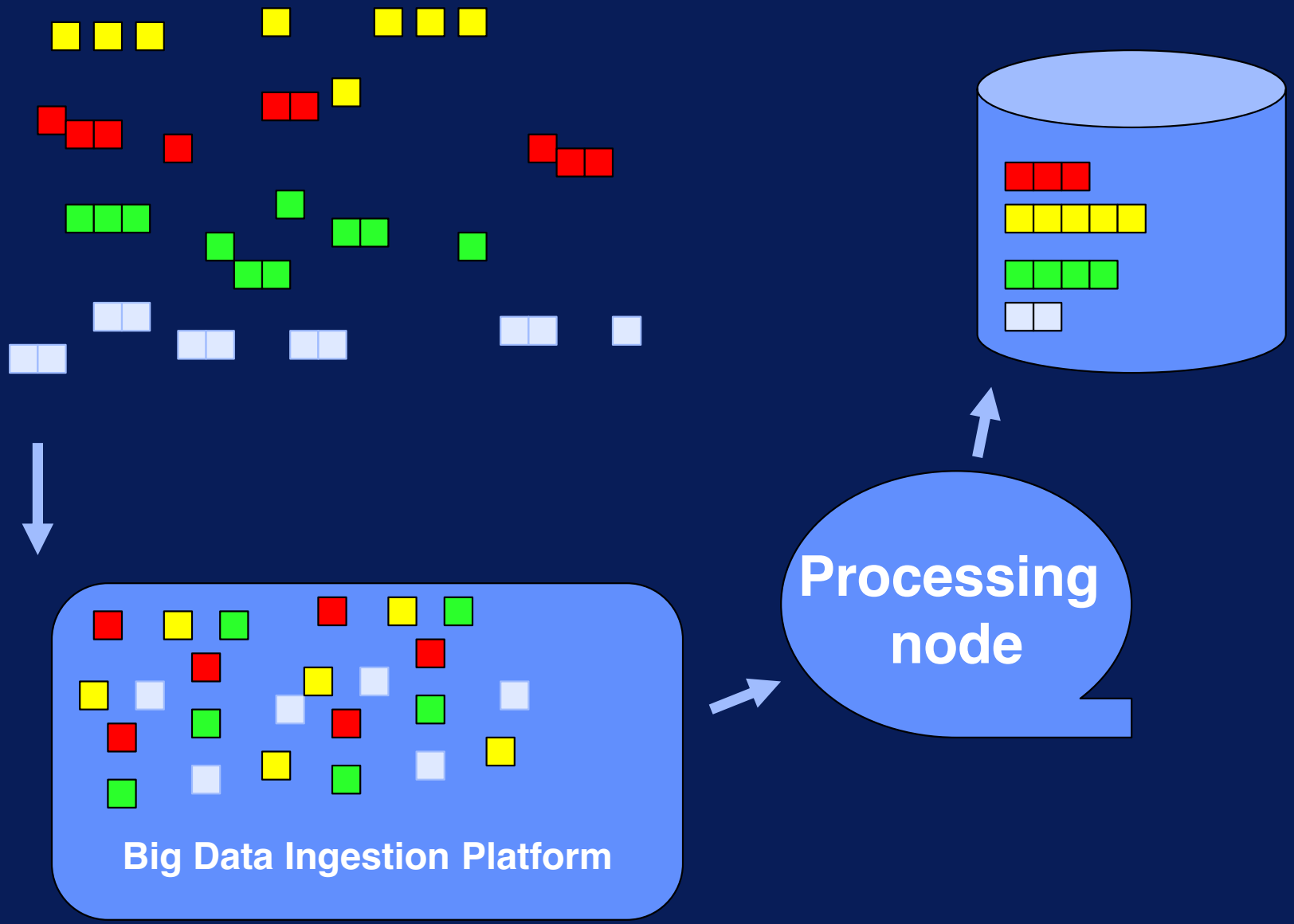


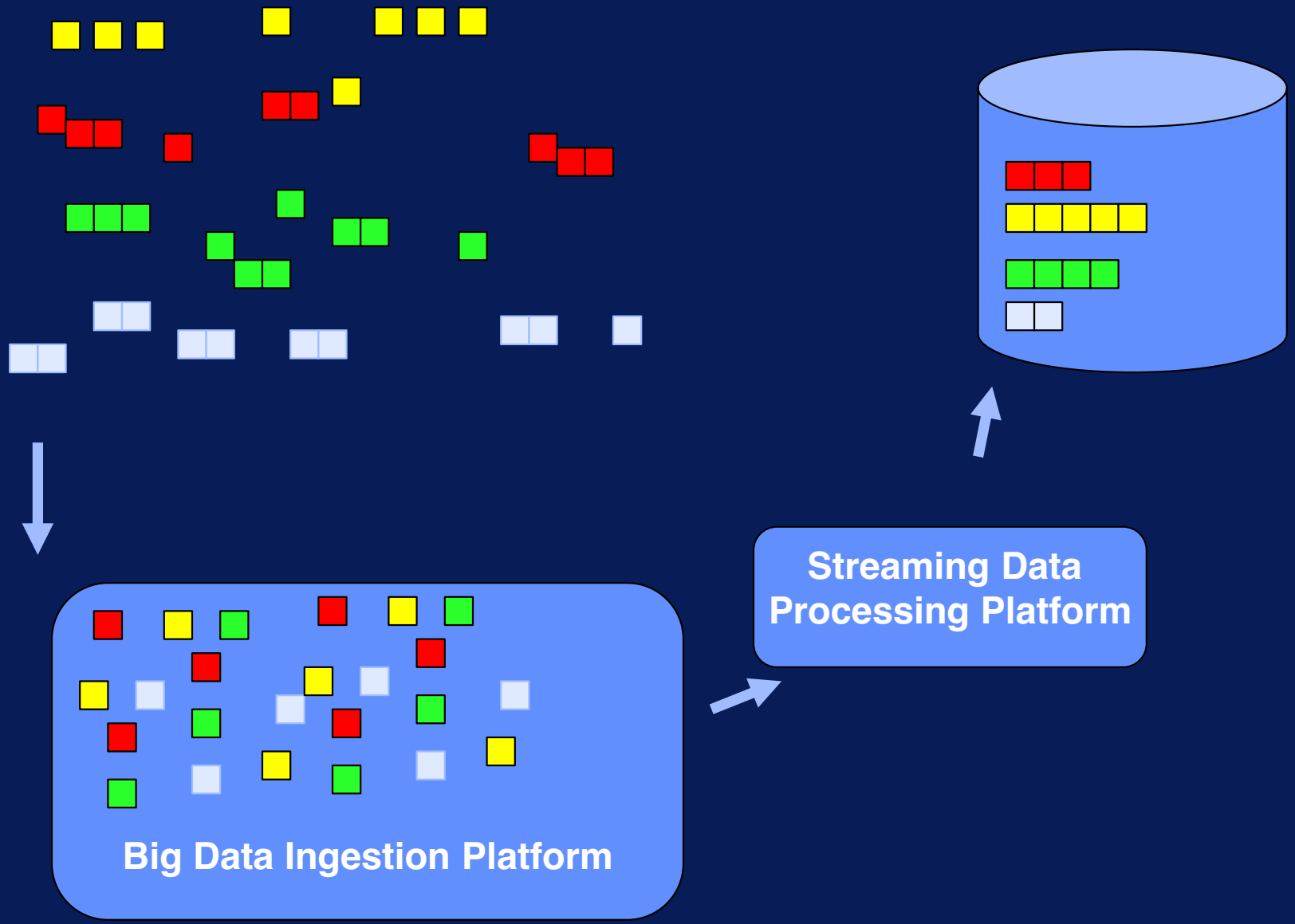
Parallelization over the input

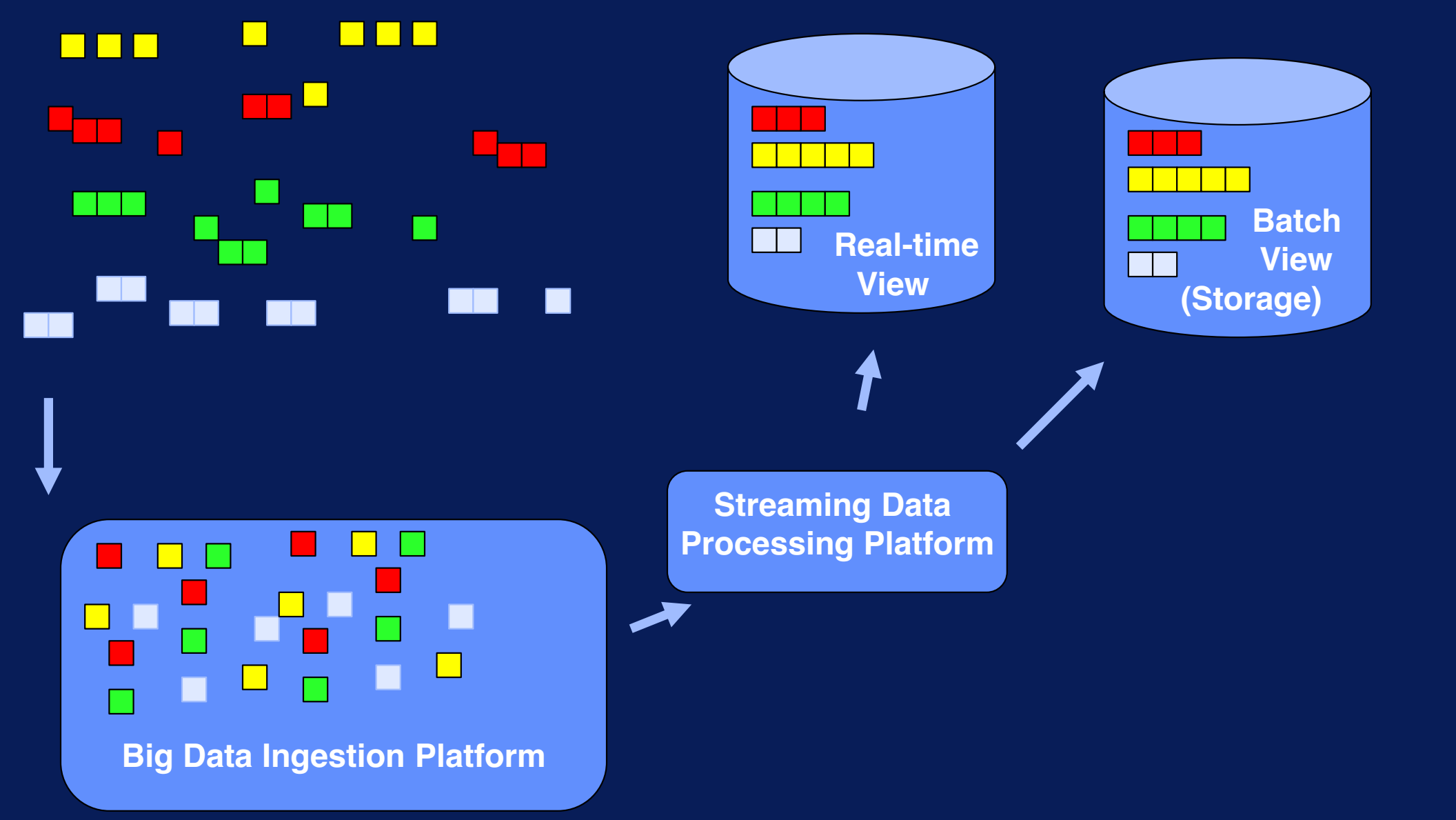
Parallelization over intermediate data

Parallelization over data groups


Processing node
platform – microbatches
STREAM PROCESSING
data ingestion


Big Data Ingestion Platform
Processing node


Streaming Data
Processing Platform
Big Data Ingestion Platform


Real-time
View
Batch
View
(Storage)
Streaming Data
Processing Platform
Big Data Ingestion Platform

big data pipelines get created to process data through an aggregated set of steps that can be represented with the split → do → merge with data parallel scalability