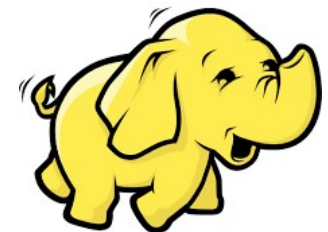


Partitioners

Tushar B. Kute,
<http://tusharkute.com>



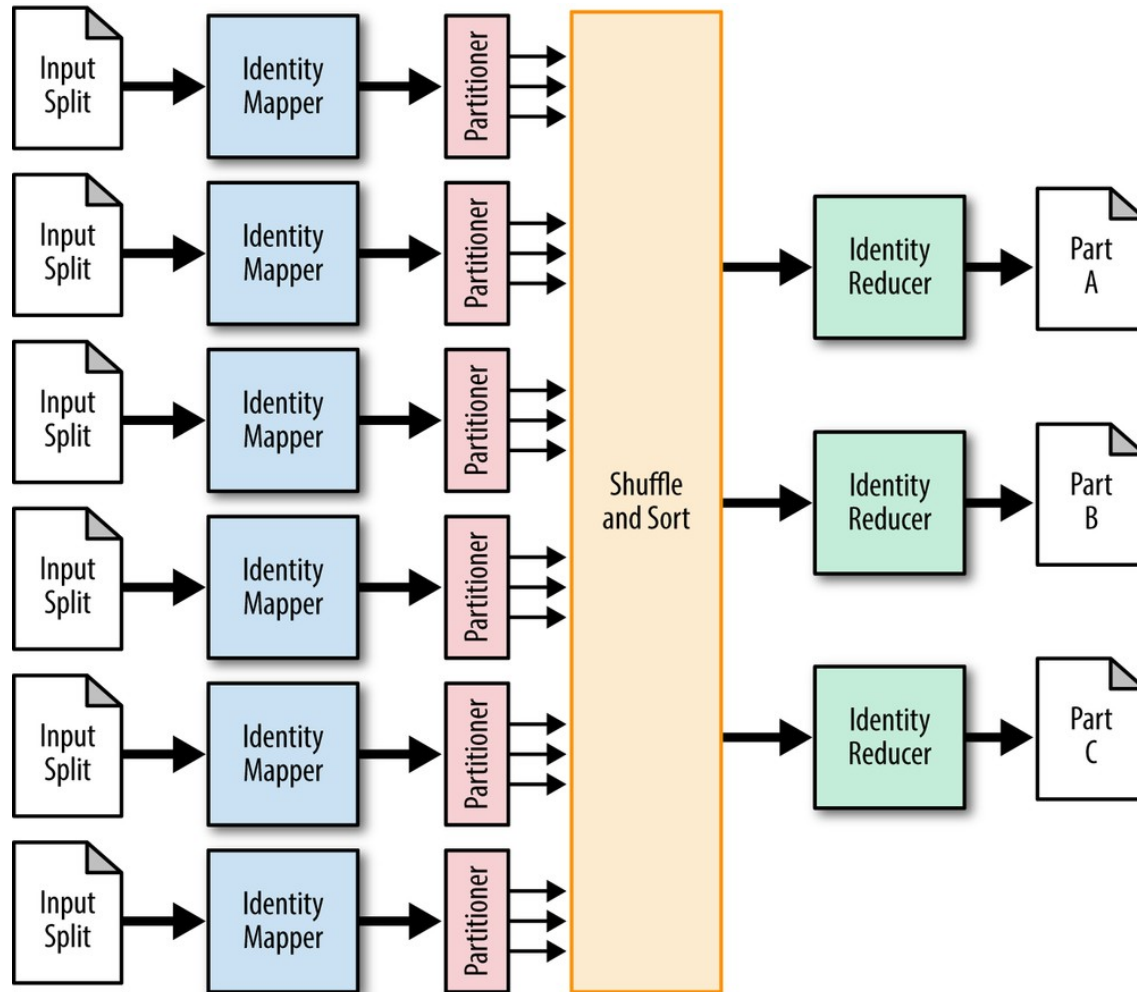
Partitioner

- A partitioner works like a condition in processing an input dataset. The partition phase takes place after the Map phase and before the Reduce phase.
- The number of partitioners is equal to the number of reducers.
- That means a partitioner will divide the data according to the number of reducers.
- Therefore, the data passed from a single partitioner is processed by a single Reducer.

Partitioner

- A partitioner partitions the key-value pairs of intermediate Map-outputs.
- It partitions the data using a user-defined condition, which works like a hash function.
- The total number of partitions is same as the number of Reducer tasks for the job.
- Let us take an example to understand how the partitioner works.

Partitioner



Input

1201	gopal	45	Male	50000
1202	manisha	40	Female	51000
1203	khaleel	34	Male	30000
1204	prasanth	30	Male	31000
1205	kiran	20	Male	40000
1206	laxmi	25	Female	35000
1207	bhavya	20	Female	15000
1208	reshma	19	Female	14000
1209	kranthi	22	Male	22000
1210	Satish	24	Male	25000
1211	Krishna	25	Male	26000
1212	Arshad	28	Male	20000
1213	lavanya	18	Female	8000

Namenode

- The map task accepts the key-value pairs as input while we have the text data in a text file. The input for this map task is as follows –
- Input – The key would be a pattern such as “any special key + filename + line number” (example: key = @input1) and the value would be the data in that line (example: value = 1201 \t gopal \t 45 \t Male \t 50000).
- Method – The operation of this map task is as follows –
 - Read the value (record data), which comes as input value from the argument list in a string.
 - Using the split function, separate the gender and store in a string variable.

Partition Task

- The partitioner task accepts the key-value pairs from the map task as its input. Partition implies dividing the data into segments.
- According to the given conditional criteria of partitions, the input key-value paired data can be divided into three parts based on the age criteria.
- Input – The whole data in a collection of key-value pairs.
 - key = Gender field value in the record.
 - value = Whole record data value of that gender.
- Method – The process of partition logic runs as follows.
 - Read the age field value from the input key-value pair.

Partition Task

```
if(age<=20)
{
    return 0;
}
else if(age>20 && age<=30)
{
    return 1 % numReduceTasks;
}
else
{
    return 2 % numReduceTasks;
}
```


Reduce Task

- The number of partitioner tasks is equal to the number of reducer tasks. Here we have three partitioner tasks and hence we have three Reducer tasks to be executed.
- Input – The Reducer will execute three times with different collection of key-value pairs.
 - key = gender field value in the record.
 - value = the whole record data of that gender.
- Method – The following logic will be applied on each collection.
 - Read the Salary field value of each record.

Thank you

This presentation is created using LibreOffice Impress 5.1.6.2, can be used freely as per GNU General Public License



@mitu_skillologies



/miTuSkillologies



@mitu_group



/company/mitu-
skillologies



MITUSkillologies

Web Resources

<https://mitu.co.in>
<http://tusharkute.com>

contact@mitu.co.in
tushar@tusharkute.com