Transcript name: MapReduce – Part 4 – Fundamental data types

English

The data that flows into and out of the mappers and reducers takes a specific form.

Data enters Hadoop in unstructured form but before it gets to the first mapper, Hadoop has changed it into key-value pairs with Hadoop supplying its own key.

The mapper produces a list of key value pairs. Both the key and the value may change from the k1 and v1 that came in to a k2 and v2. There can now be duplicate keys coming out of the mappers. The shuffle step will take care of grouping them together.

The output of the shuffle is the input to the reducer step. Now, we still have a list of the v2's that come out of the mapper step, but they are grouped by their keys and there is no longer more than one record with the same key.

Finally, coming out of the reducer is, potentially, an entirely new key and value, k3 and v3. For example, if your reducer summed the values associated with each k2, your k3 would be equal to k2 and your v3 would be the sum of the list of v2s.

Let us look at an example of a simple data flow. Say we want to transform the input on the left to the output on the right. On the left, we just have letters. On the right, we have counts of the number of occurrences of each letter in the input.

Hadoop does the first step for us. It turns the input data into key-value pairs and supplies its own key: an increasing sequence number.

The function we write for the mapper needs to take these key-value pairs and produce something that the reduce step can use to count occurrences. The simplest solution is make each letter a key and make every value a 1.

The shuffle groups records having the same key together, so we see B now has two values, both 1, associated with it.

The reduce is simple: it just sums the values it is given to produce a sum for each key.

This lesson is continued in the next video.