MapReduce

Map-Reduce is a scalable programming model that simplifies distributed processing of data. Map-Reduce consists of three main steps: Mapping, Shuffling and Reducing. An easy way to think about a Map-Reduce job is to compare it with act of 'delegating' a large task to a group of people, and then combining the result of each person's effort, to produce the final outcome.

Let's take an example to bring the point across. You just heard about this great news at your office, and are throwing a party for all your colleagues! You decide to cook Pasta for the dinner. Four of your friends, who like cooking, also volunteer to join you in preparation. The task of preparing Pasta broadly involves chopping the vegetables, cooking, and garnishing.

Let's take the job of chopping the vegetables and see how it is analogous to a map-reduce task. Here the raw vegetables are symbolic of the input data, your friends are equivalent to compute nodes, and final chopped vegetables are analogous to desired outcome. Each friend is allotted onions, tomatoes and peppers to chop and weigh.

You would also like to know how much of each vegetable types you have in the kitchen. You would also like to chop these vegetables while this calculation is occurring. In the end, the onions should be in one large bowl with a label that displays its weight in pounds, tomatoes in a separate one, and so on.

MAP

To start with, you assign each of your four friends a random mix of different types of vegetables.

They are required to use their 'compute' powers to chop them and measure the weight of each type of veggie. They need to ensure not to mix different types of veggies. So each friend will generate a mapping of <key, value> pairs that looks like:

Friend X:

<tomatoes, 5 lbs>

<onions, 10 lbs>

<garlic, 2 lbs>

Friend Y:

<onions, 22 lbs>

<green peppers, 5 lbs>

...

Seems like you are having a really big party! Now that your friends have chopped the vegetables, and labeled each bowl with the weight and type of vegetable, we move to the next stage: Shuffling.

SHUFFLE: This stage is also called Grouping. Here you want to group the veggies by their types. You assign different parts of your kitchen to each type of veggie, and your friends are supposed to group the bowls, so that like items are placed together:

North End of Kitchen:

<tomatoes, 5 lbs>

<tomatoes, 11 lbs>

West End of Kitchen:

<onions, 10 lbs>

<onions, 22 lbs>

<onions, 1.4 lbs>

East End of Kitchen:

<green peppers, 3 lbs>

<green peppers, 10 lbs>

The party starts in a couple of hours, but you are impressed by what your friends have accomplished by Mapping and Grouping so far! The kitchen looks much more organized now and the raw material is chopped. The final stage of this task is to measure how much of each veggie you actually have. This brings us to the Reduce stage.

REDUCE

In this stage, you ask each of your friend to collect items of same type, put them in a large bowl, and label this large bowl with sum of individual bowl weights. Your friends cannot wait for the party to start, and immediately start 'reducing' small bowls. In the end, you have nice large bowls, with total weight of each vegetable labeled on it.

<tomatoes, 36 lbs>

<green peppers, 15lbs>

<onions, 44 lbs>

The number represents total weight of that vegetable after reducing from smaller bowls

Your friends ('compute nodes') just performed a Map-Reduce task to help you get started with cooking the Pasta. Since you were coordinating the entire exercise, you are "The Master" node of this Map-Reduce task. Each of your friends took roles of Mappers, Groupers and Reducers at different times. This example demonstrates the power of this technique.

This simple and powerful technique can be scaled very easily if more of your friends decide to join you. In future, we will continue to add more articles on different open source tools that will help you easily implement Map-Reduce to solve your computational problems.

Source: http://words.sdsc.edu/words-data-science/mapreduce

✓ Complete