MapReduce

The Story of Sam

Adapted from SALSA HPC Group
http://salsahpc.indiana.edu

Sam's Mother

- Believed "an apple a day keeps a doctor away"

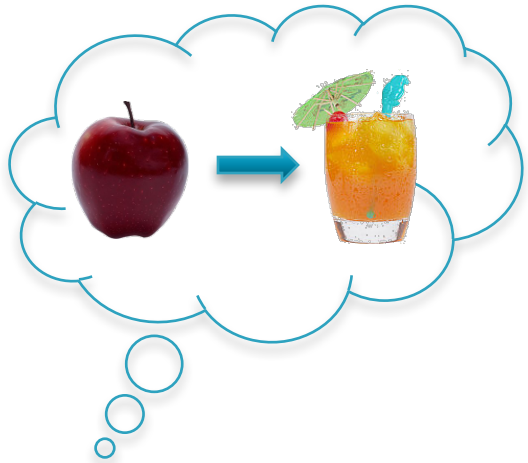
Mother

An Apple

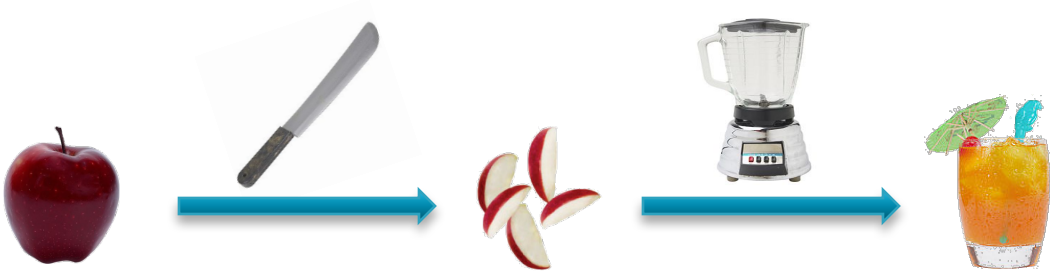
Sam

One day

- Sam thought of “drinking” the apple



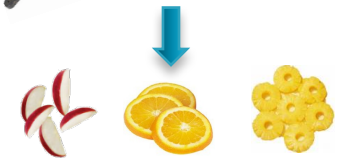
- He used a  to cut the  and a  to make juice.



Next Day

- Sam applied his invention to all the fruits he could find in the *fruit basket*

- (map  ‘(  ))



- (reduce  ‘(  ))

A *list of values* mapped into another *list of values*, which gets reduced into a *single value*

Classical Notion of MapReduce in Functional Programming

18 Years Later

- Sam got his first job in JuiceRUs for his talent in making juice

Wait!

- Now, it's not just one basket but a whole *container* of fruits

- Also, they produce a *list* of juice types separately

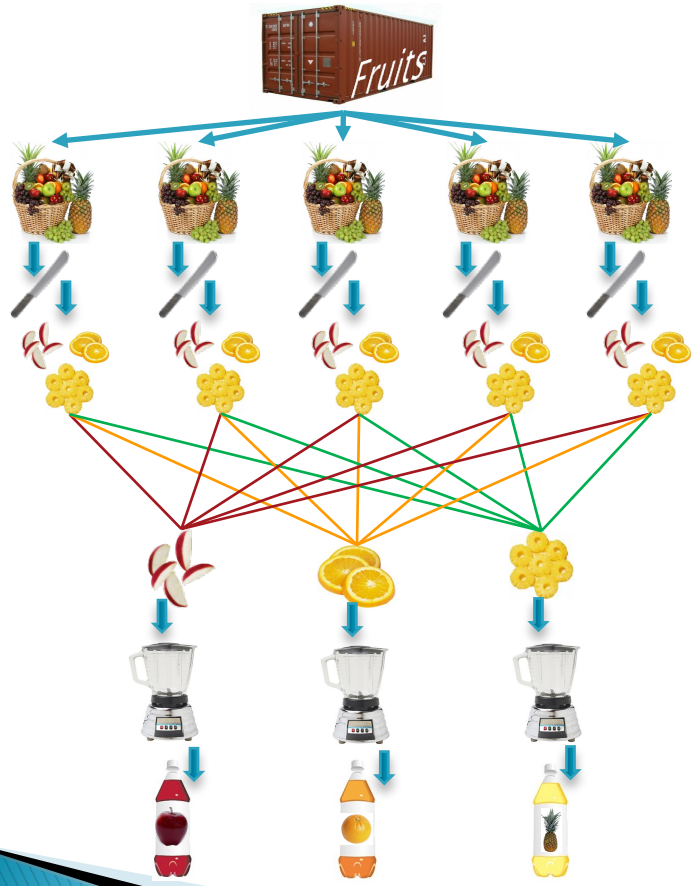
Large data and list of values for output

- But, Sam had just ONE  and ONE 

NOT ENOUGH !!

Brave Sam

- Implemented a *parallel* version of his innovation



Each input to a map is a list of <key, value> pairs

(<a🍎> , <o🍊> , <p🍍> , …)

A *list of <key, value>* pairs mapped into another *list of <key, value>* pairs which gets grouped by the key and reduced into a *list of values*

Each output of a map is a list of <key, value> pairs

(<a’,🍎🍎> , <o’,🍊🍊> , <p’,🍍🍍> , …)

Grouped by key

Each [illegible] st>
(pos [illegible] e
grouping/hashing mechanism)
e.g. <a’, (🍎🍎🍎 …)>

Reduced into a list of values

The idea of MapReduce in Data Intensive Computing

Afterwards

- Sam realized,
  - To create his favorite mix fruit juice he can use a *combiner* after the mappers.
  - If several <key, value-list> fall into the same group (based on the grouping/hashing algorithm) then use the blender (reducer) separately on each of them
  - The knife (mapper) and blender (reducer) should not contain residue after use – *Side Effect Free*
  - In general reducer should be *associative* and *commutative*

That's All Folks!

- We think Sam was you ☺