Big Data using Hadoop and Map-reduce

Ken Lau

April 9th, 2015

What is Big Data?

- Data that is too large to be processed on a single machine
- Examples:
  - Amazon/Netflix data
    - Pages viewed and how long you stayed
  - Sports
    - Tracking ticket sales
  - Twitter stream data
- **Problem:** Very large data sets in general
- Side Note: Feel free to ask questions at any time

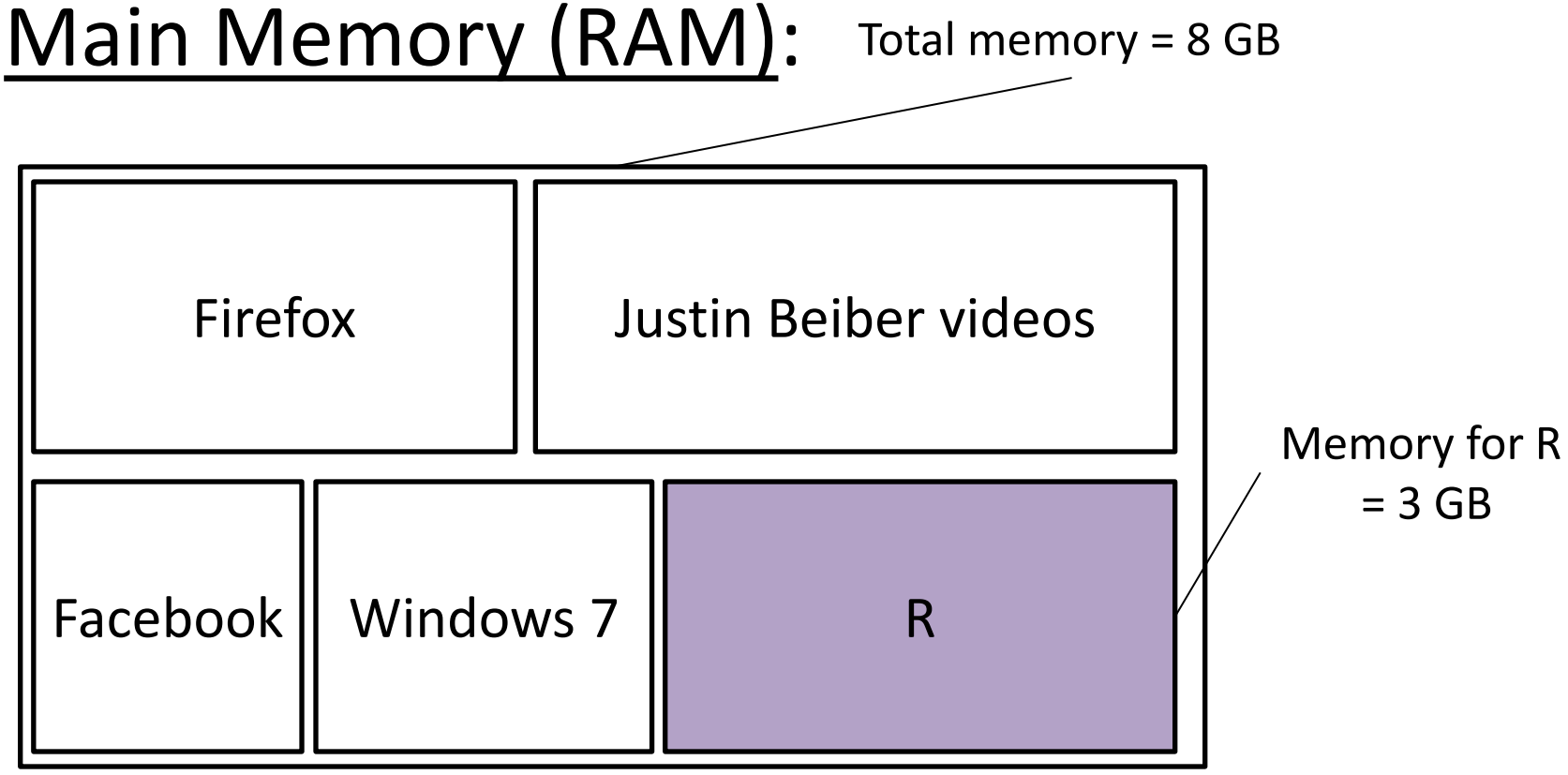
Twitter Stream Data


```
{
    "created_at": "Thu Jul 10 05:59:10 +0000 2014",
    "id": 487113782989045760,
    "id_str": "487113782989045760",
    "text": "@_nickisthebomb_ @mace_krause @therealtmoo it's a clothing brand and yes tmo, justin beiber is alive.
    "source": "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitter for iPhone</a>",
    "truncated": false,
    "in_reply_to_status_id": 487113583839281150,
    "in_reply_to_status_id_str": "487113583839281152",
    "in_reply_to_user_id": 1105410138,
    "in_reply_to_user_id_str": "1105410138",
    "in_reply_to_screen_name": "_nickisthebomb_",
    "user": {
        "id": 599803691,
        "id_str": "599803691",
        "name": "your mom ",
        "screen_name": "micahevangaline",
        "location": "",
        "url": null,
        "description": "P.N.P H.E.B S.N.G & i like mini vans & booty too strong in the fam bam",
```

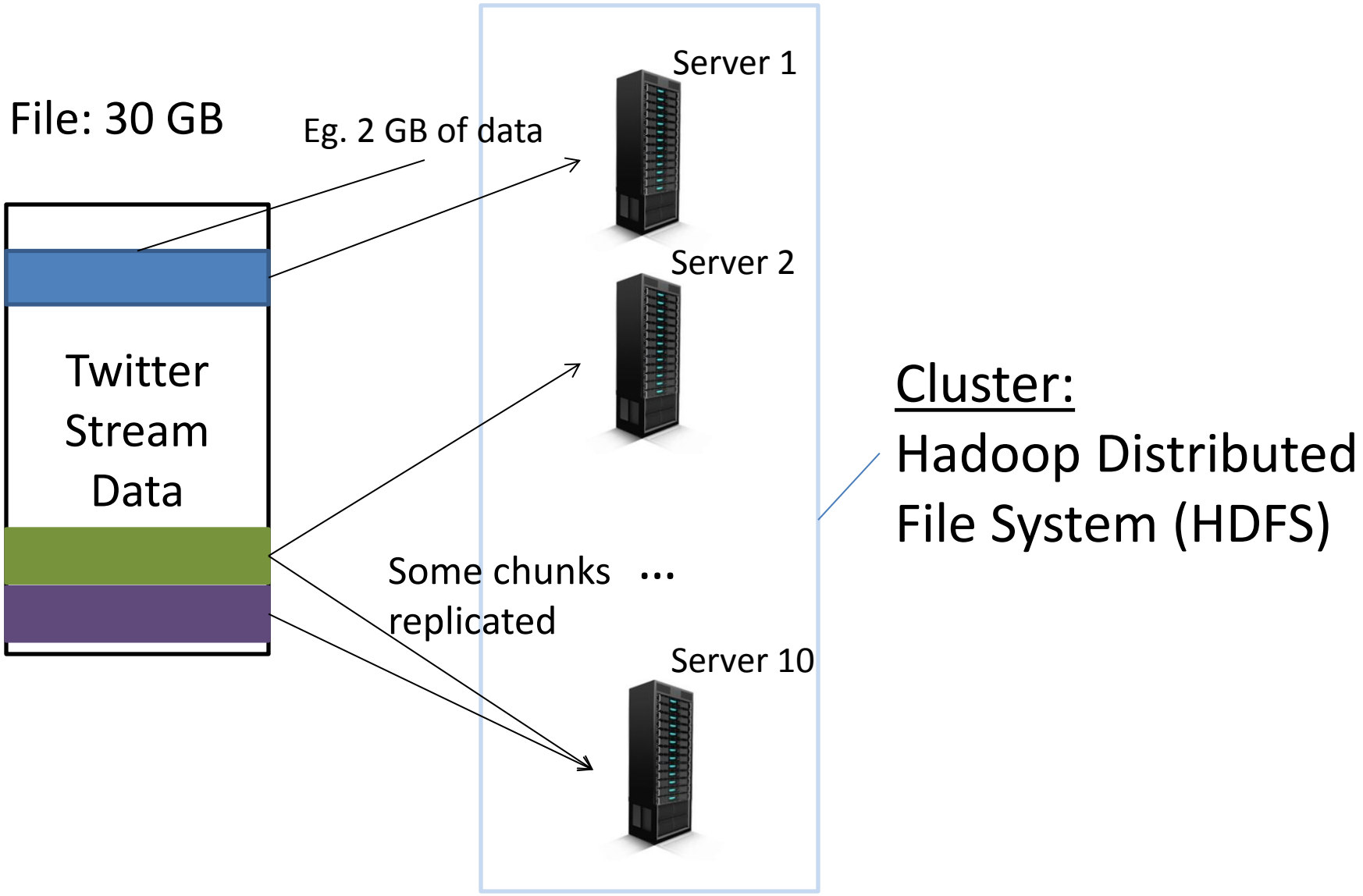

“it's a clothing brand and yes tmo, justin beiber is alive. sadly.”

How much space do we have for R?

Main Memory (RAM):



How to maintain large data sets?



What is Hadoop?

- Hadoop and map-reduce provides a programming framework to process data that can't fit into main memory.

- Hadoop is written in Java, but the Hadoop Streaming API lets you write your programs in Python/R for simpler tasks.

- Amazon Web Service (AWS) provides a suite of cloud computing services that are paid by the hour.

Example

- Task:
  - Compute the word frequencies of the stream of twitter data. Example in the next slide.
- Typical Python/R Solution:
  - Extract the text key of each input stream.
  - Split the text by blank spaces.
  - Maintain a running count of the words using a hash table with a [word -> count] mapping.
    - Use a dictionary in Python
    - Use a list in R

```
{
    "created_at": "Thu Jul 10 05:59:10 +0000 2014",
    "id": 487113782989045760,
    "id_str": "487113782989045760",
    "text": "@_nickisthebomb_ @mace_krause @therealtmoo it's a clothing brand and yes tmo, justin beiber is alive.
    "source": "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitter for iPhone</a>",
    "truncated": false,
    "in_reply_to_status_id": 487113583839281150,
    "in_reply_to_status_id_str": "487113583839281152",
    "in_reply_to_user_id": 1105410138,
    "in_reply_to_user_id_str": "1105410138",
    "in_reply_to_screen_name": "_nickisthebomb_",
    "user": {
        "id": 599803691,
        "id_str": "599803691",
        "name": "your mom ",
        "screen_name": "micahevangaline",
        "location": "",
        "url": null,
        "description": "P.N.P H.E.B S.N.G & i like mini vans & booty too strong in the fam bam",
```


{…, it's: 3, a: 5, clothing: 2, ….., justin: 3, beiber: 1,…}

Increase each word by 1 each time you see it appear again.

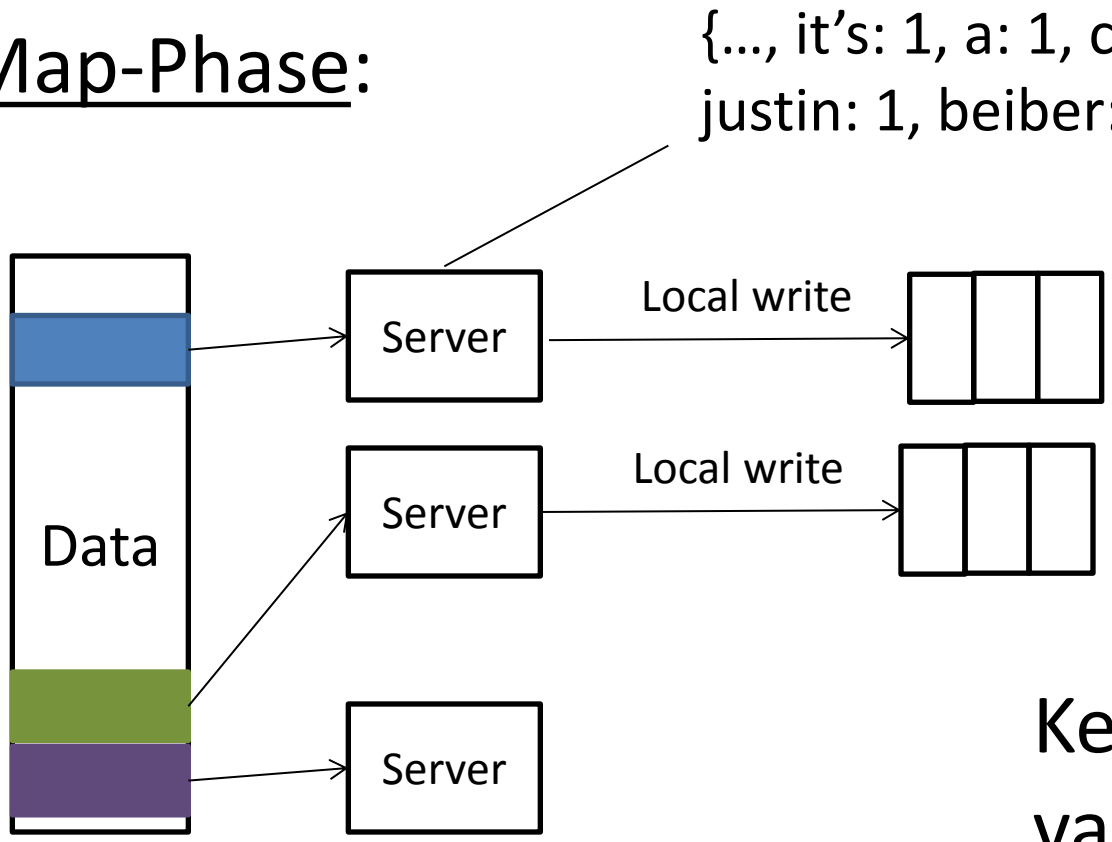
How to implement map-reduce to solve the same problem

- There are 3 phases in general:
  - Map-phase
  - Shuffle-phase
  - Reduce-phase

- Example execution on the data stream in the next slide.

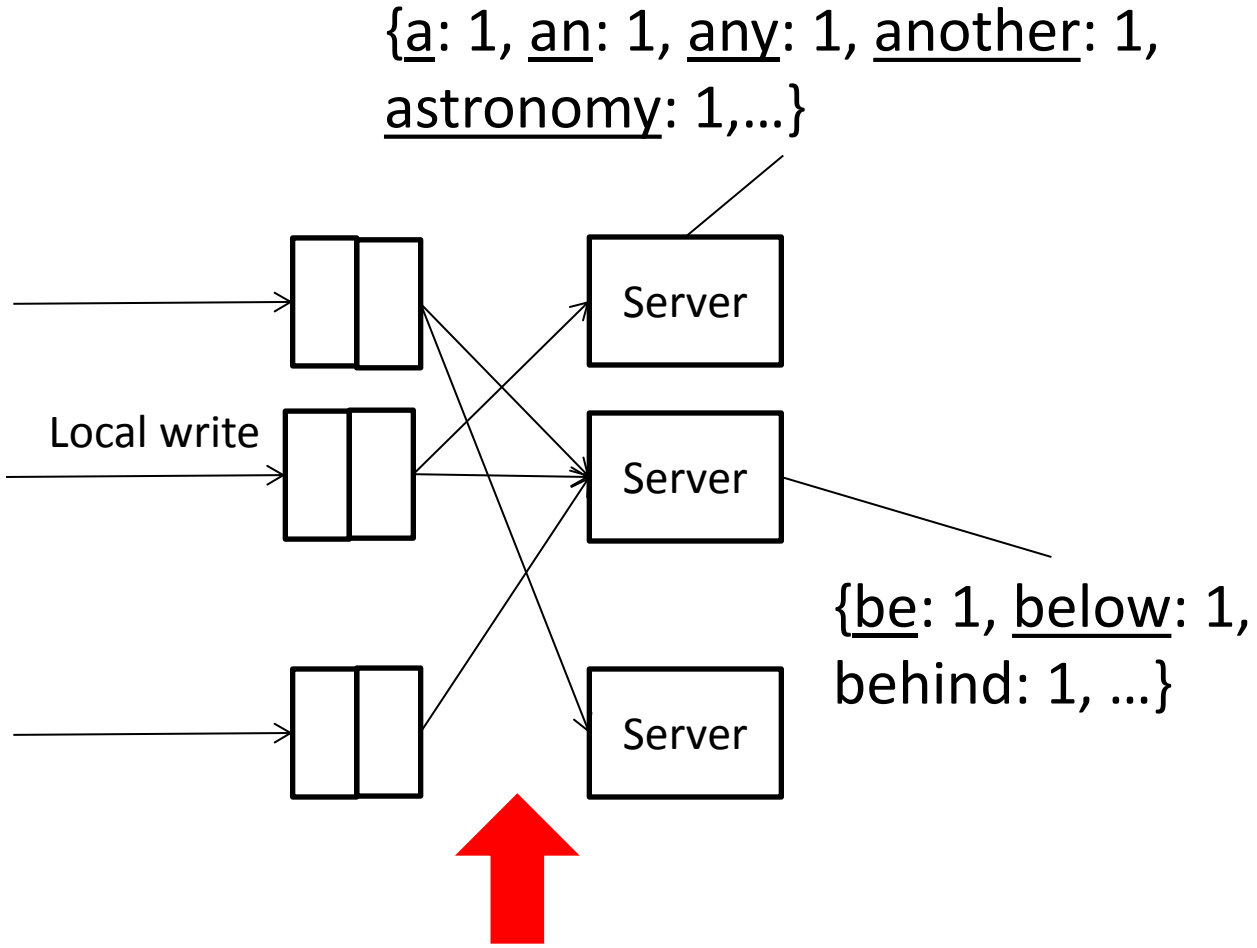
Text: **"it's a clothing brand and yes tmo, justin beiber is alive. sadly."**

Map-Phase:



Key value pair with value always 1. (<word>: 1)

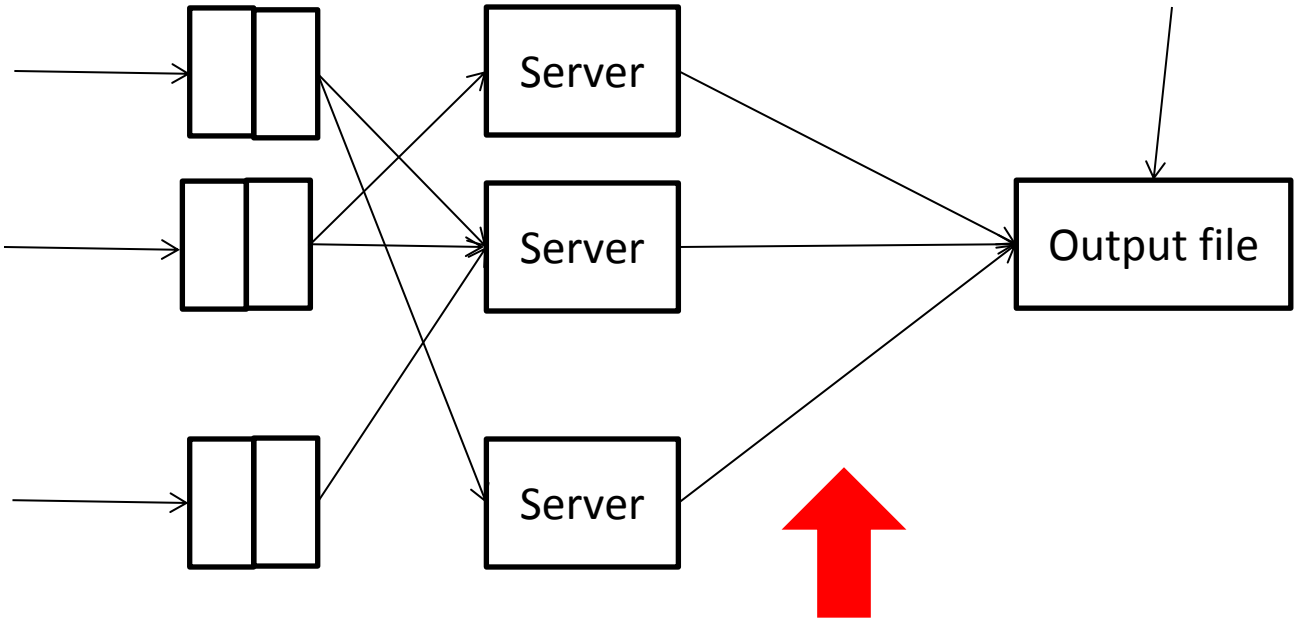
Shuffle-Phase:



Reduce-Phase:

Count up the occurrences as in the original solution

{a: 20, an: 5, any: 10, another: 3, astronomy: 4,…}



References:

- “MapReduce: Simplified Data Processing on Large Clusters”
  - Jeffrey Dean and Sanjay Ghemawat

- Wikipedia/Blogs