Big Data Systems

Djellel Difallah

Spring 2023

Lecture 9 – Apache Hadoop 1.0

MapReduce

Outline

- High order functions (in Python)
- MapReduce Distributed Computation
- Hadoop 1.0

High order functions in Python

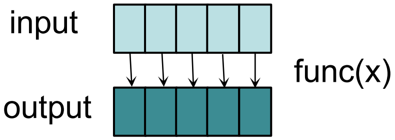
MAP(), FILTER(), REDUCE() FUNCTIONS

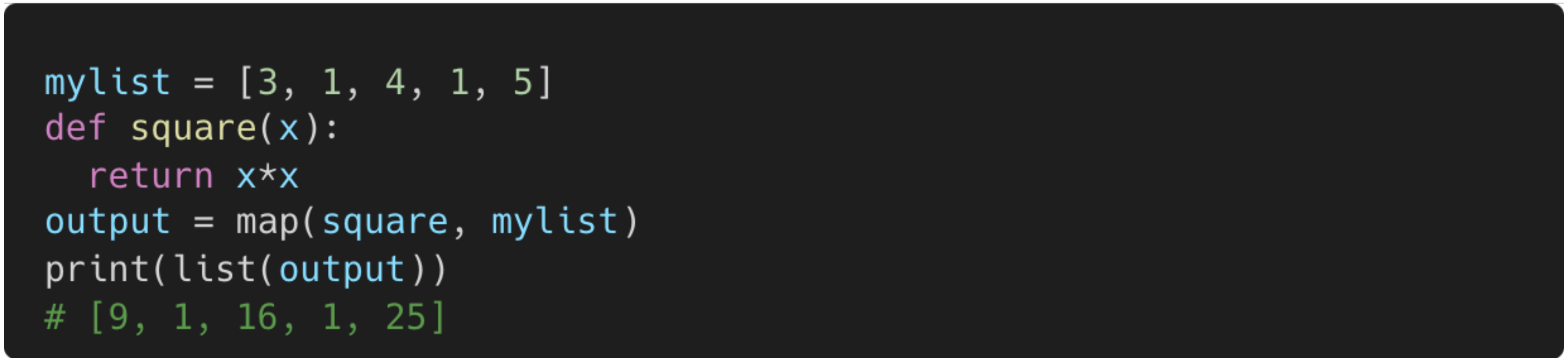
High-Order Functions in Python

- A higher-order function is a function that takes another function as a parameter
  - function of a function
- Functional programming paradigm
  - Frequently used in *big* data processing: successive functions composition, avoiding changing state and mutable data (lazy evaluation)
- Built-in High-Order Functions in Python:
  - **map()**
  - **filter()**
  - **reduce(): part of the functools library**
- Lambda functions (one liner, single use function)
  - lambda arguments: expression

Map() function

- The map function applies a given function to each item in of an iterable object (e.g., list, tuple)
  - map(function, iterable)
  - Operates element-wise on the input iterable

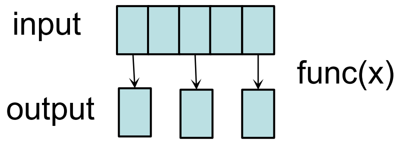


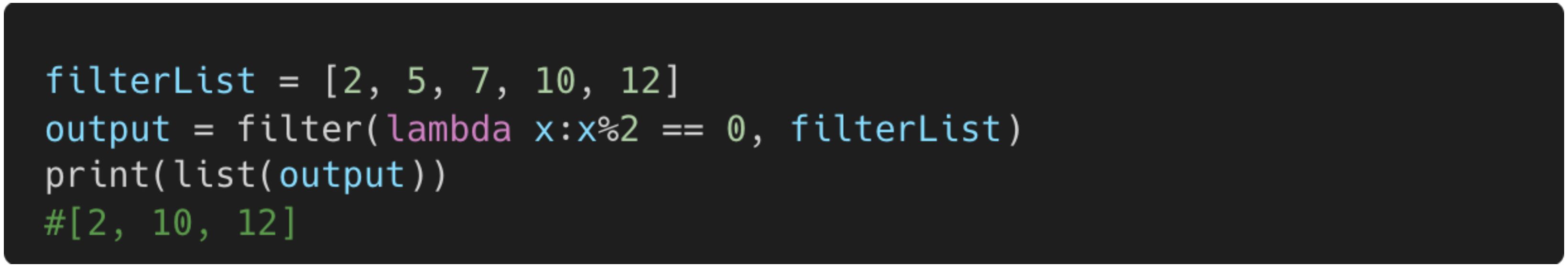


```
mylist = [3, 1, 4, 1, 5]
def square(x):
  return x*x
output = map(square, mylist)
print(list(output))
# [9, 1, 16, 1, 25]
```

Filter() function

- Filters elements in an iterable object based on a provided boolean function
  - filter(function, iterable)
  - Keep elements for which the function returns True
  - Example using lambda function:

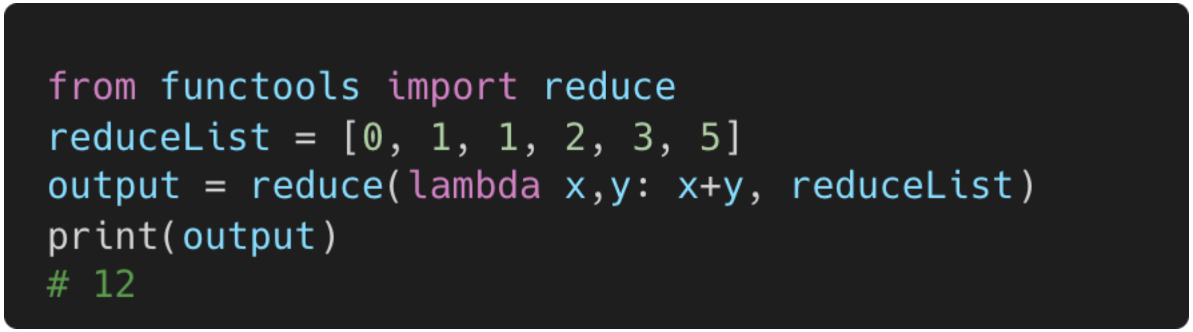




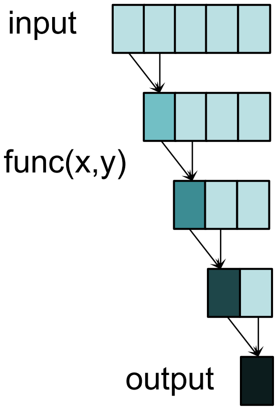
```
filterList = [2, 5, 7, 10, 12]
output = filter(lambda x:x%2 == 0, filterList)
print(list(output))
#[2, 10, 12]
```

Reduce() function

- Applies a function to elements of an iterable object successively reducing them to a single value
  - reduce(function, iterable, initializer)
  - need to: `import functools`
  - The reduce function must accept two arguments and return a single value.

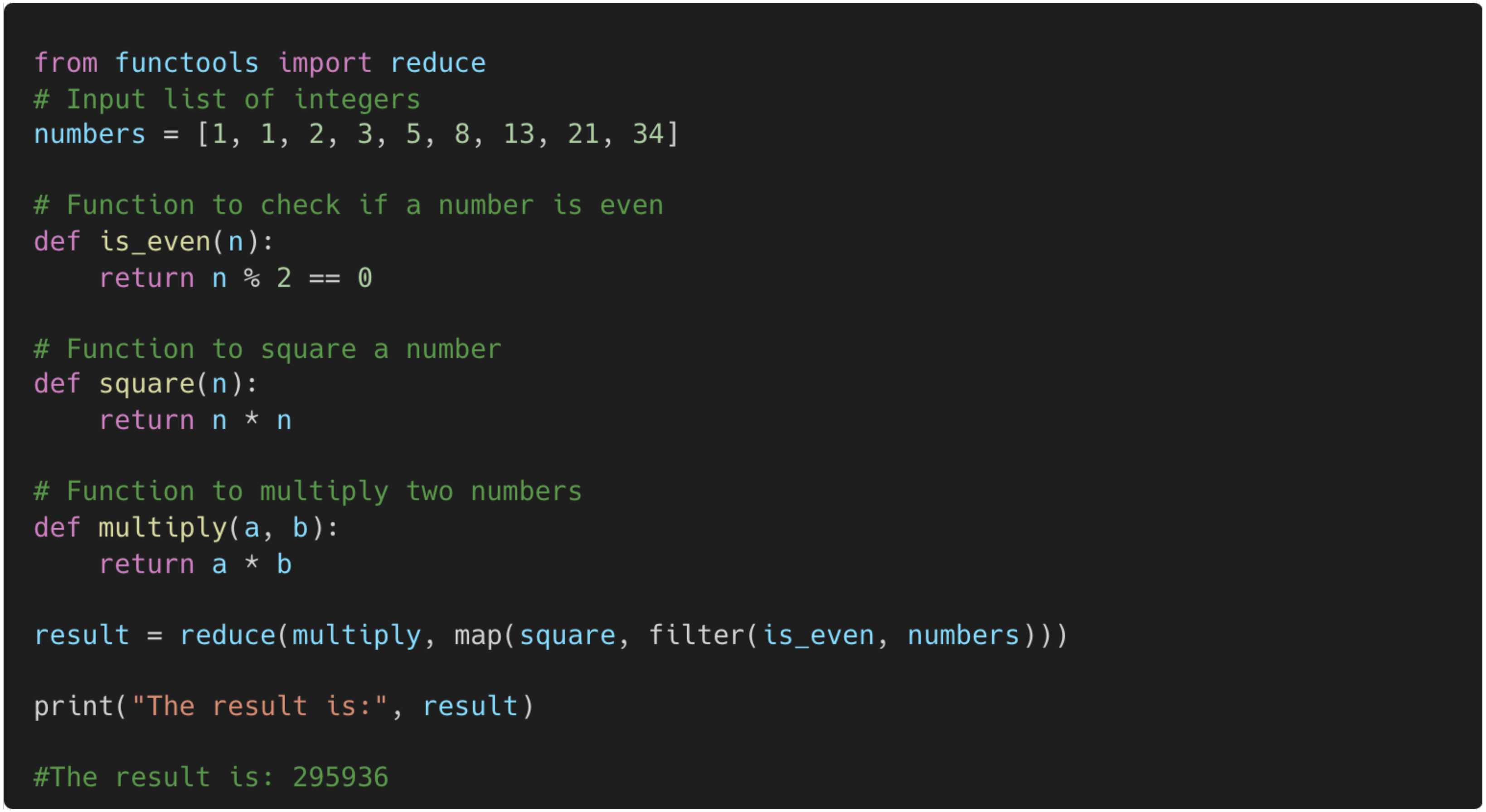


```python
from functools import reduce
reduceList = [0, 1, 1, 2, 3, 5]
output = reduce(lambda x,y: x+y, reduceList)
print(output)
# 12
```



Composition

- Product of the squares of even numbers in a given list of integers.



```
from functools import reduce
# Input list of integers
numbers = [1, 1, 2, 3, 5, 8, 13, 21, 34]

# Function to check if a number is even
def is_even(n):
    return n % 2 == 0

# Function to square a number
def square(n):
    return n * n

# Function to multiply two numbers
def multiply(a, b):
    return a * b

result = reduce(multiply, map(square, filter(is_even, numbers)))

print("The result is:", result)

#The result is: 295936
```

Distributed Programming Paradigm

MAPREDUCE

MapReduce Programming Model

- A framework for processing huge datasets on certain kinds of distributable problems
- The programmer needs to write two functions:
  - **Map**(Key, Value)
    - Output: <KEY1, VALUE1>

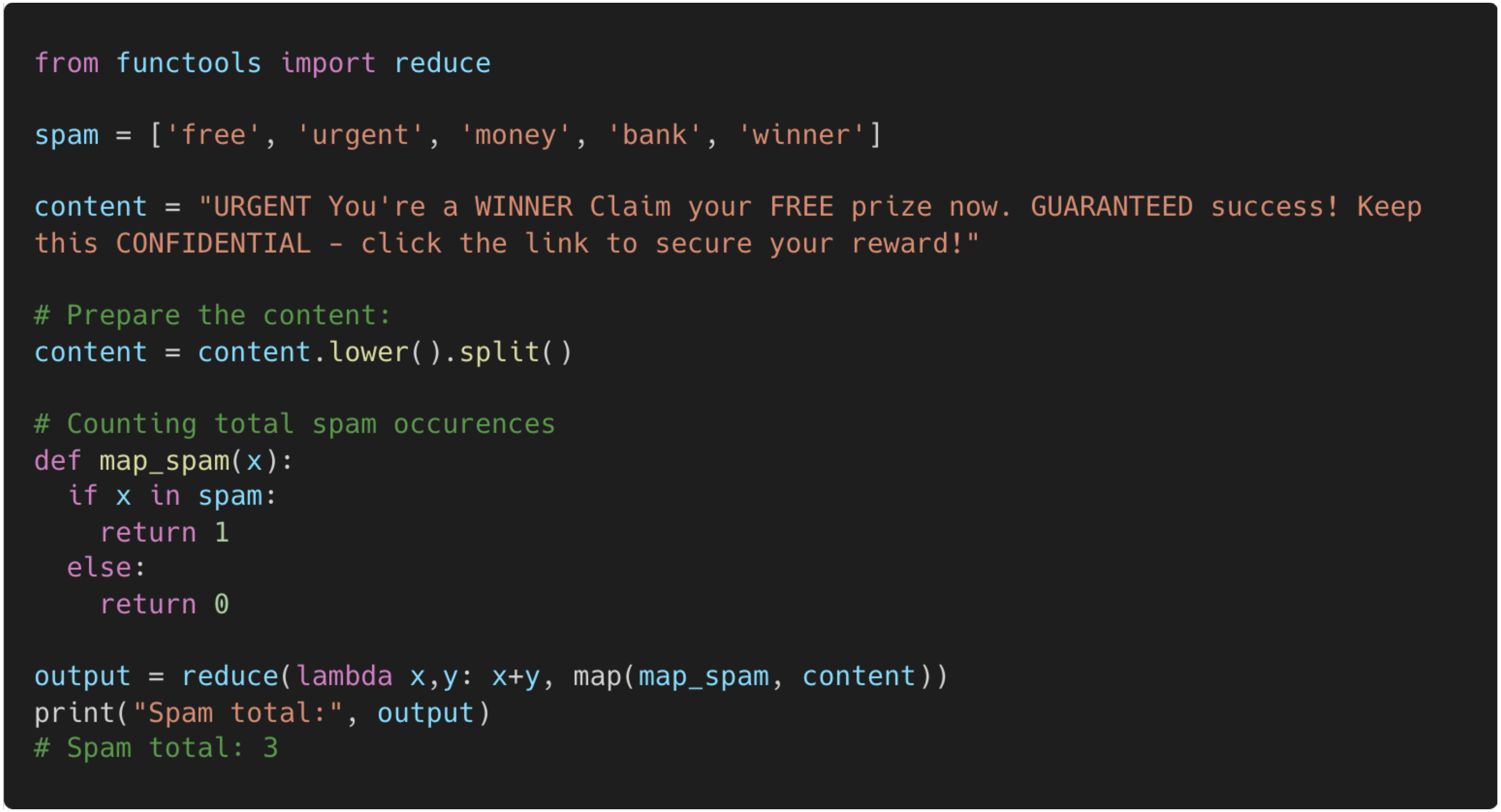
  All values with the same key are then sent the same reducer

  - **Reduce**(KEY1, [List of VALUE1])
    - Output: <KEY2, VALUE2>

  Working examples:

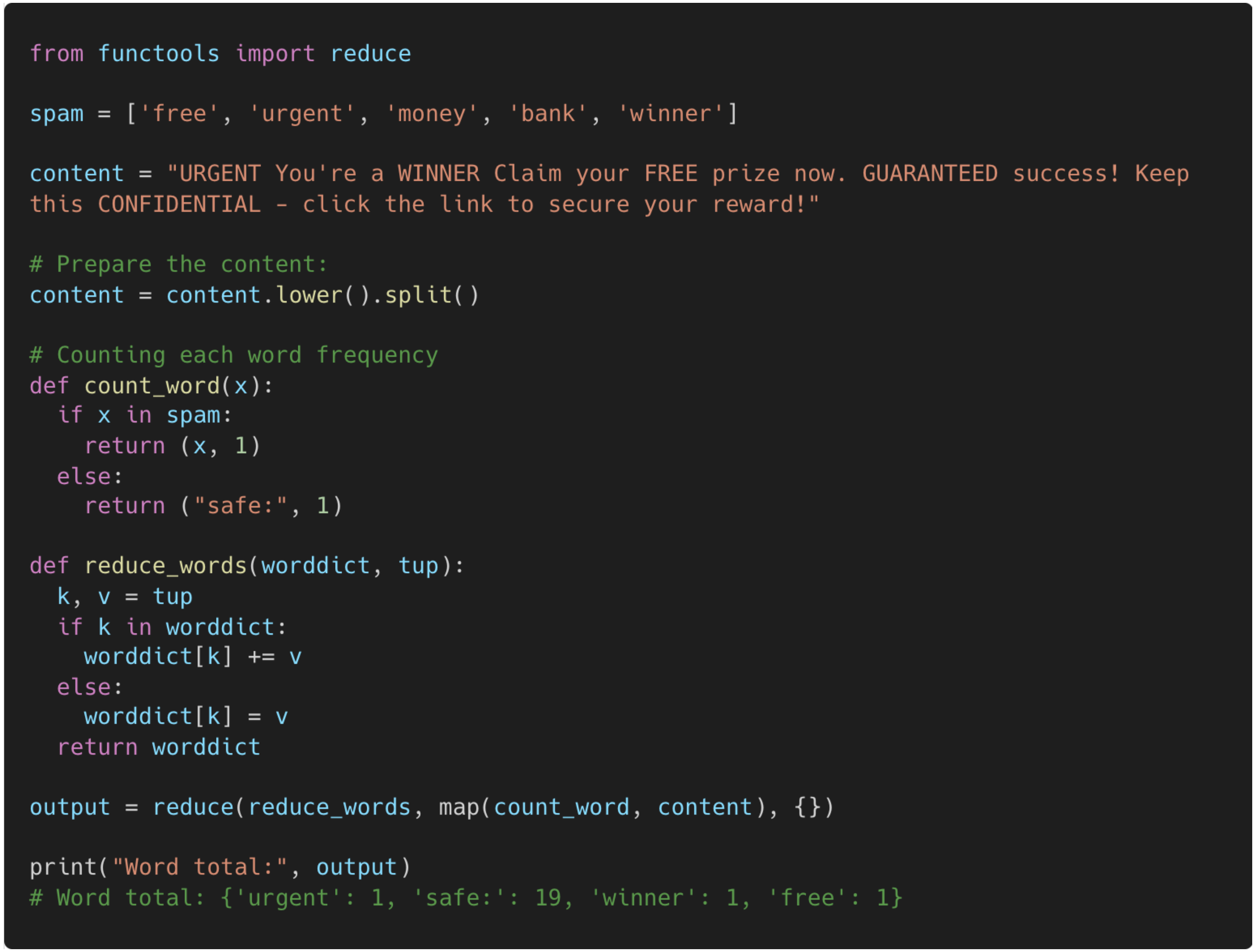
  - Word Count
  - Word Length Count

Frequency of spam words



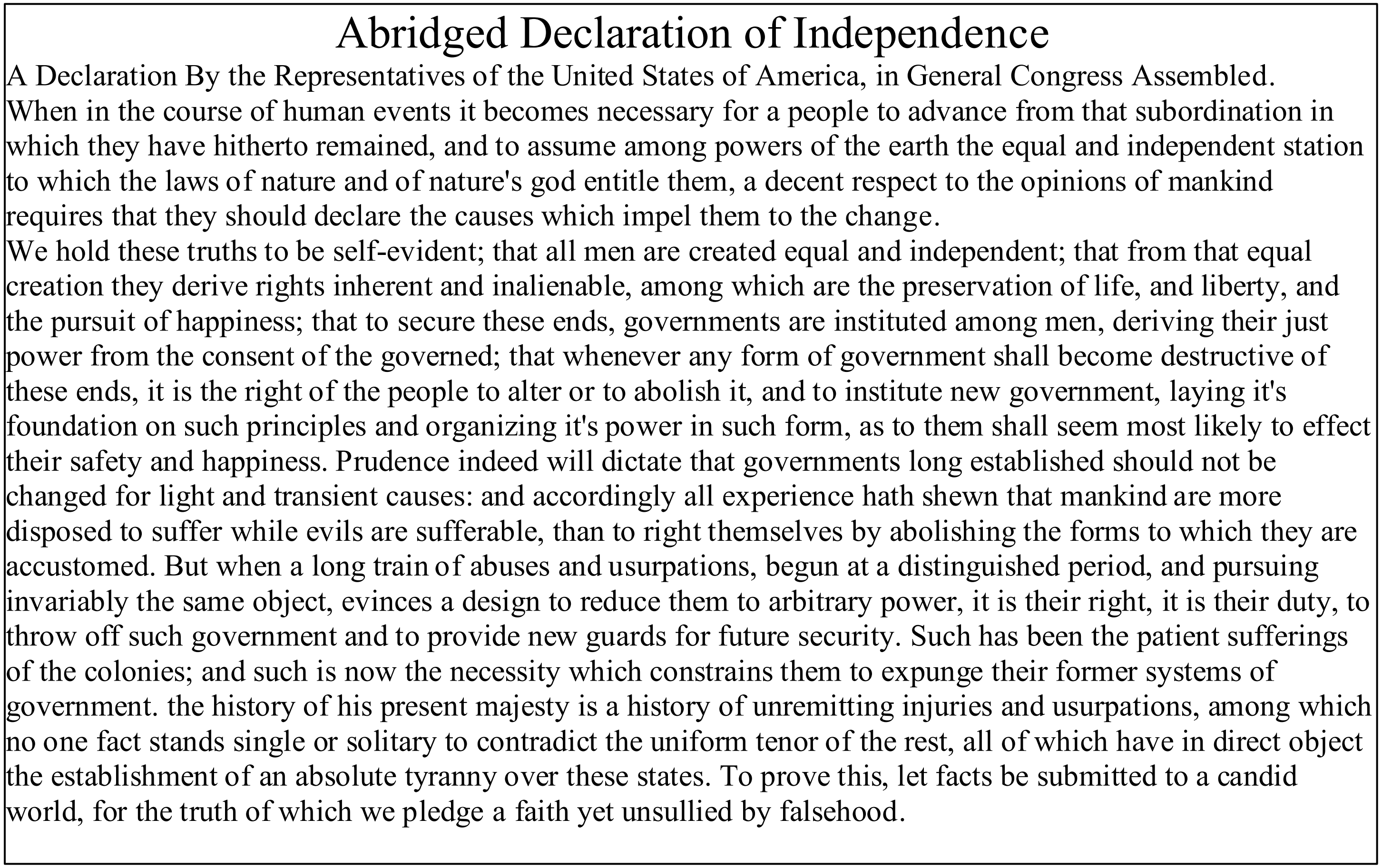
```python
from functools import reduce

spam = ['free', 'urgent', 'money', 'bank', 'winner']

content = "URGENT You're a WINNER Claim your FREE prize now. GUARANTEED success! Keep
this CONFIDENTIAL - click the link to secure your reward!"

# Prepare the content:
content = content.lower().split()

# Counting total spam occurences
def map_spam(x):
  if x in spam:
    return 1
  else:
    return 0

output = reduce(lambda x,y: x+y, map(map_spam, content))
print("Spam total:", output)
# Spam total: 3
```

Word Count



```
from functools import reduce

spam = ['free', 'urgent', 'money', 'bank', 'winner']

content = "URGENT You're a WINNER Claim your FREE prize now. GUARANTEED success! Keep
this CONFIDENTIAL - click the link to secure your reward!"

# Prepare the content:
content = content.lower().split()

# Counting each word frequency
def count_word(x):
  if x in spam:
    return (x, 1)
  else:
    return ("safe:", 1)

def reduce_words(worddict, tup):
  k, v = tup
  if k in worddict:
    worddict[k] += v
  else:
    worddict[k] = v
  return worddict

output = reduce(reduce_words, map(count_word, content), {})

print("Word total:", output)
# Word total: {'urgent': 1, 'safe:': 19, 'winner': 1, 'free': 1}
```

Word Length Count



Abridged Declaration of Independence

A Declaration By the Representatives of the United States of America, in General Congress Assembled.

When in the course of human events it becomes necessary for a people to advance from that subordination in which they have hitherto remained, and to assume among powers of the earth the equal and independent station to which the laws of nature and of nature's god entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the change.

We hold these truths to be self-evident; that all men are created equal and independent; that from that equal creation they derive rights inherent and inalienable, among which are the preservation of life, and liberty, and the pursuit of happiness; that to secure these ends, governments are instituted among men, deriving their just power from the consent of the governed; that whenever any form of government shall become destructive of these ends, it is the right of the people to alter or to abolish it, and to institute new government, laying it's foundation on such principles and organizing it's power in such form, as to them shall seem most likely to effect their safety and happiness. Prudence indeed will dictate that governments long established should not be changed for light and transient causes: and accordingly all experience hath shewn that mankind are more disposed to suffer while evils are sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long train of abuses and usurpations, begun at a distinguished period, and pursuing invariably the same object, evinces a design to reduce them to arbitrary power, it is their right, it is their duty, to throw off such government and to provide new guards for future security. Such has been the patient sufferings of the colonies; and such is now the necessity which constrains them to expunge their former systems of government. the history of his present majesty is a history of unremitting injuries and usurpations, among which no one fact stands single or solitary to contradict the uniform tenor of the rest, all of which have in direct object the establishment of an absolute tyranny over these states. To prove this, let facts be submitted to a candid world, for the truth of which we pledge a faith yet unsullied by falsehood.

Word Length Count

Map Task 1
(204 words)

Abridged Declaration of Independence

A Declaration By the Representatives of the United States of America, in General Congress Assembled.

When in the course of human events it becomes necessary for a people to advance from that subordination in which they have hitherto remained, and to assume among powers of the earth the equal and independent station to which the laws of nature and of nature's god entitle them, a decent respect to the opinions of mankind requires that they should declare the causes which impel them to the change.

We hold these truths to be self-evident; that all men are created equal and independent; that from that equal creation they derive rights inherent and inalienable, among which are the preservation of life, and liberty, and the pursuit of happiness; that to secure these ends, governments are instituted among men, deriving their just power from the consent of the governed; that whenever any form of government shall become destructive of these ends, it is the right of the people to alter or to abolish it, and to institute new government, laying it's foundation on such principles and organizing it's power in such form, as to them shall seem most likely to effect their safety and happiness. Prudence indeed will

(key, value)

(yellow, 17)
(red, 77)
(blue, 107)
(pink, 3)

Yellow: 10+

Red: 5..9

Blue: 2..4

Pink: = 1

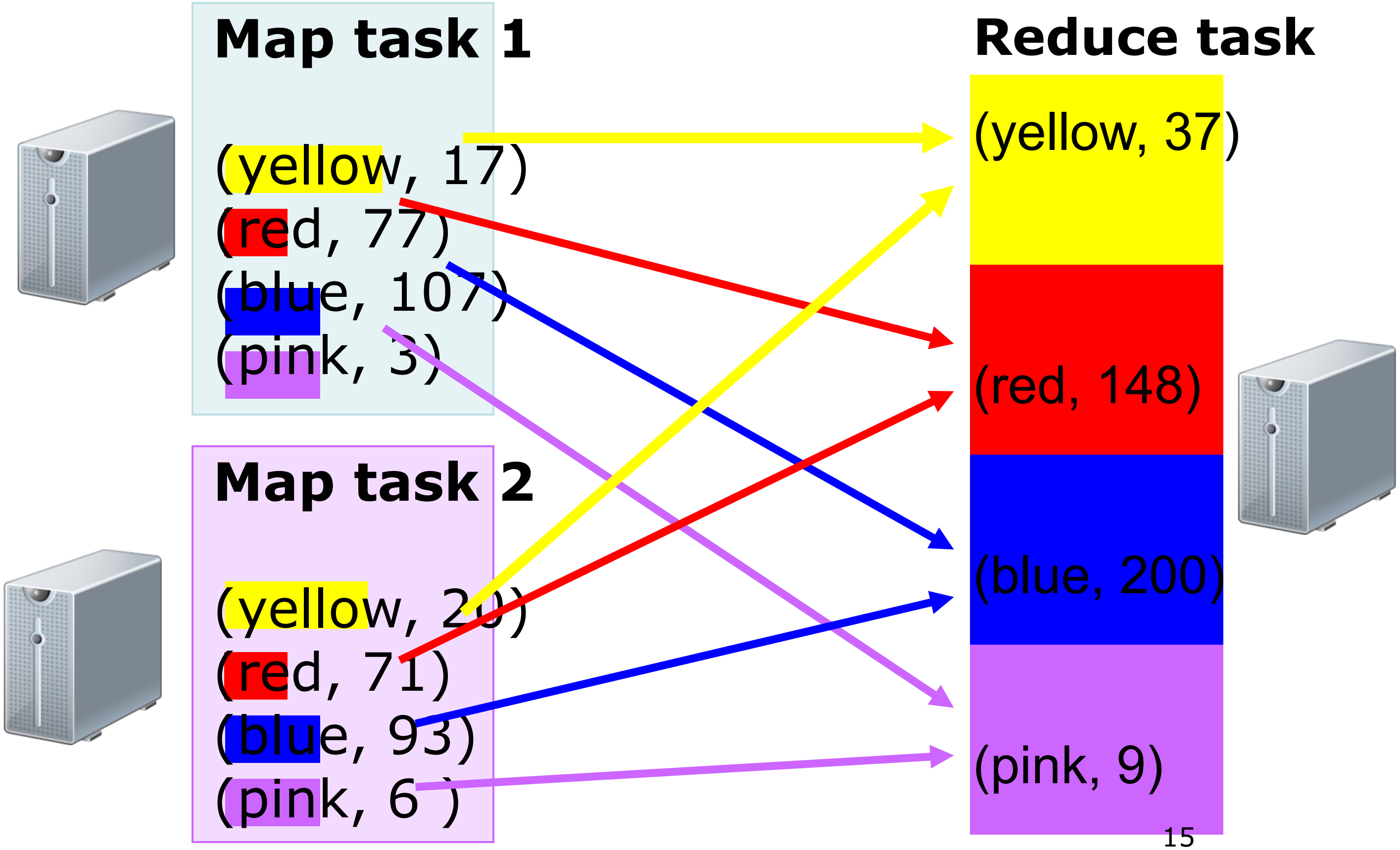
Map Task 2
(190 words)

dictate that governments long established should not be changed for light and transient causes: and accordingly all experience hath shewn that mankind are more disposed to suffer while evils are sufferable, than to right themselves by abolishing the forms to which they are accustomed. But when a long train of abuses and usurpations, begun at a distinguished period, and pursuing invariably the same object, evinces a design to reduce them to arbitrary power, it is their right, it is their duty, to throw off such government and to provide new guards for future security. Such has been the patient sufferings of the colonies; and such is now the necessity which constrains them to expunge their former systems of government. the history of his present majesty is a history of unremitting injuries and usurpations, among which no one fact stands single or solitary to contradict the uniform tenor of the rest, all of which have in direct object the establishment of an absolute tyranny over these states. To prove this, let facts be submitted to a candid world, for the truth of which we pledge a faith yet unsullied by falsehood.

(yellow, 20)
(red, 71)
(blue, 93)
(pink, 6)

Word Length Count

Map is a **GROUP BY** operation
Reduce is an **AGGREGATE** operation



Map task 1

(yellow, 17)
(red, 77)
(blue, 107)
(pink, 3)

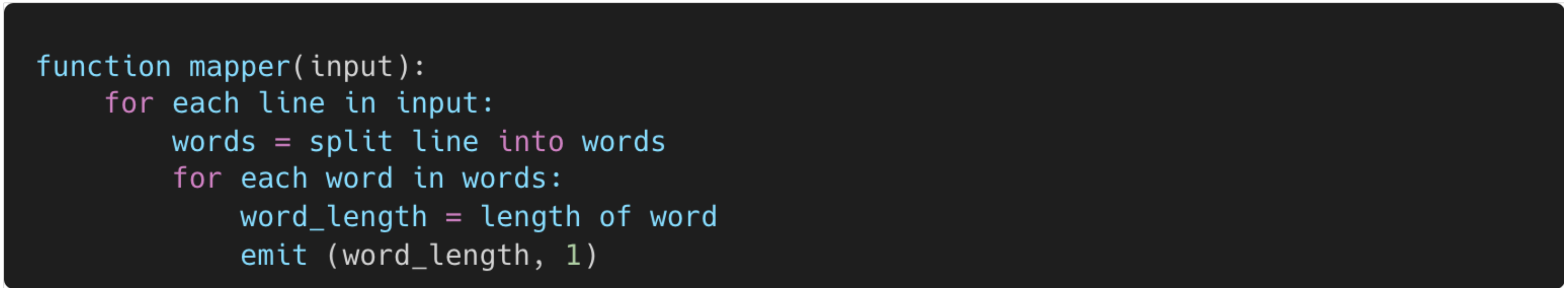
Map task 2

(yellow, 20)
(red, 71)
(blue, 93)
(pink, 6)

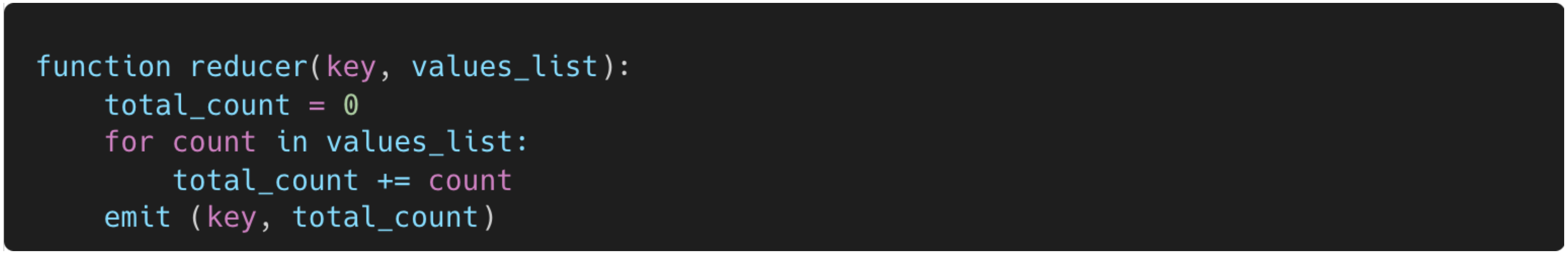
Reduce task

(yellow, 37)
(red, 148)
(blue, 200)
(pink, 9)

Word Length Count



```
function mapper(input):
    for each line in input:
        words = split line into words
        for each word in words:
            word_length = length of word
            emit (word_length, 1)
```



```
function reducer(key, values_list):
    total_count = 0
    for count in values_list:
        total_count += count
    emit (key, total_count)
```

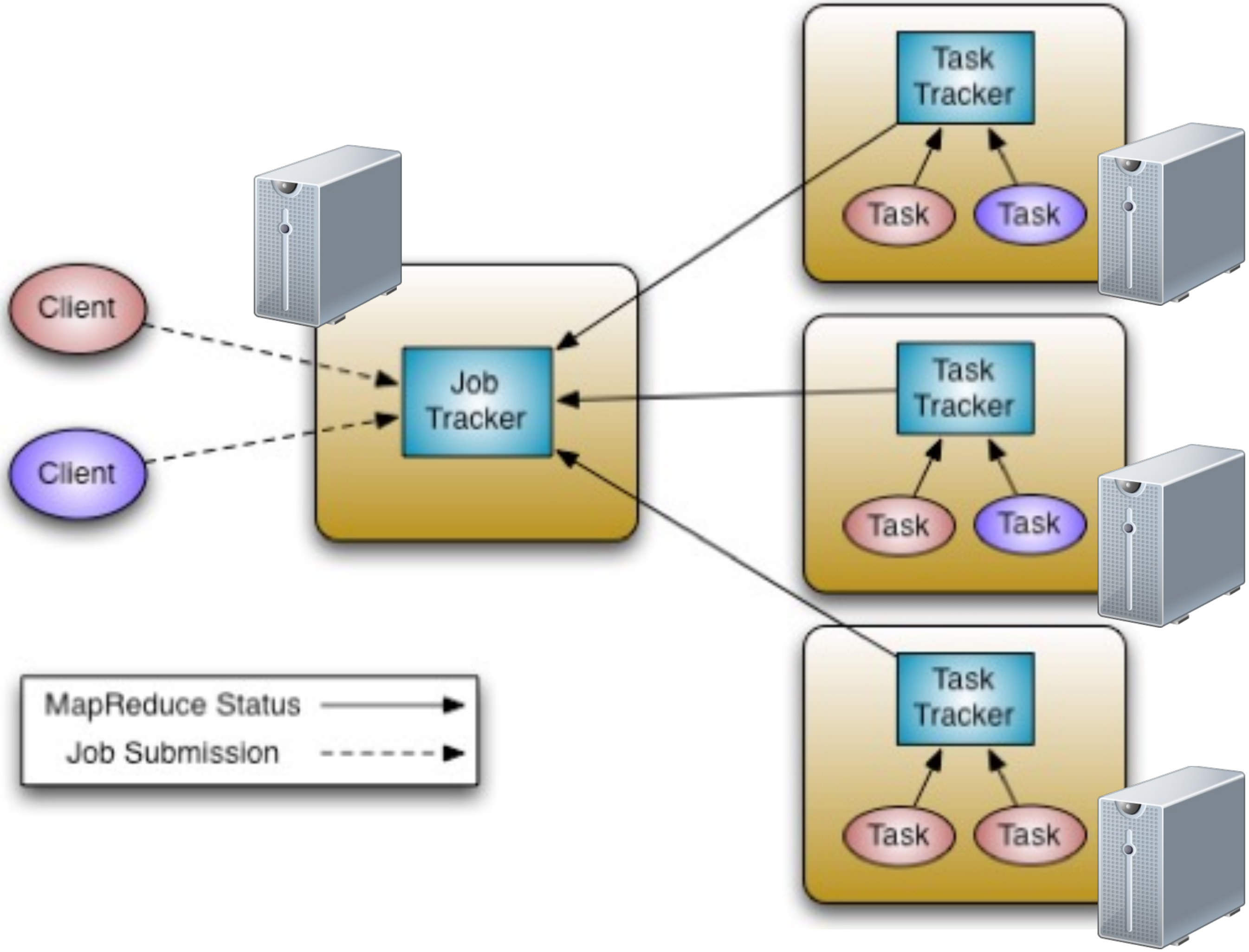
Distributed Computation

HADOOP 1.0 (MAPREDUCE)

MapReduce

- The programming model
- The execution framework
- The specific implementation

Hadoop 1.0 Architecture



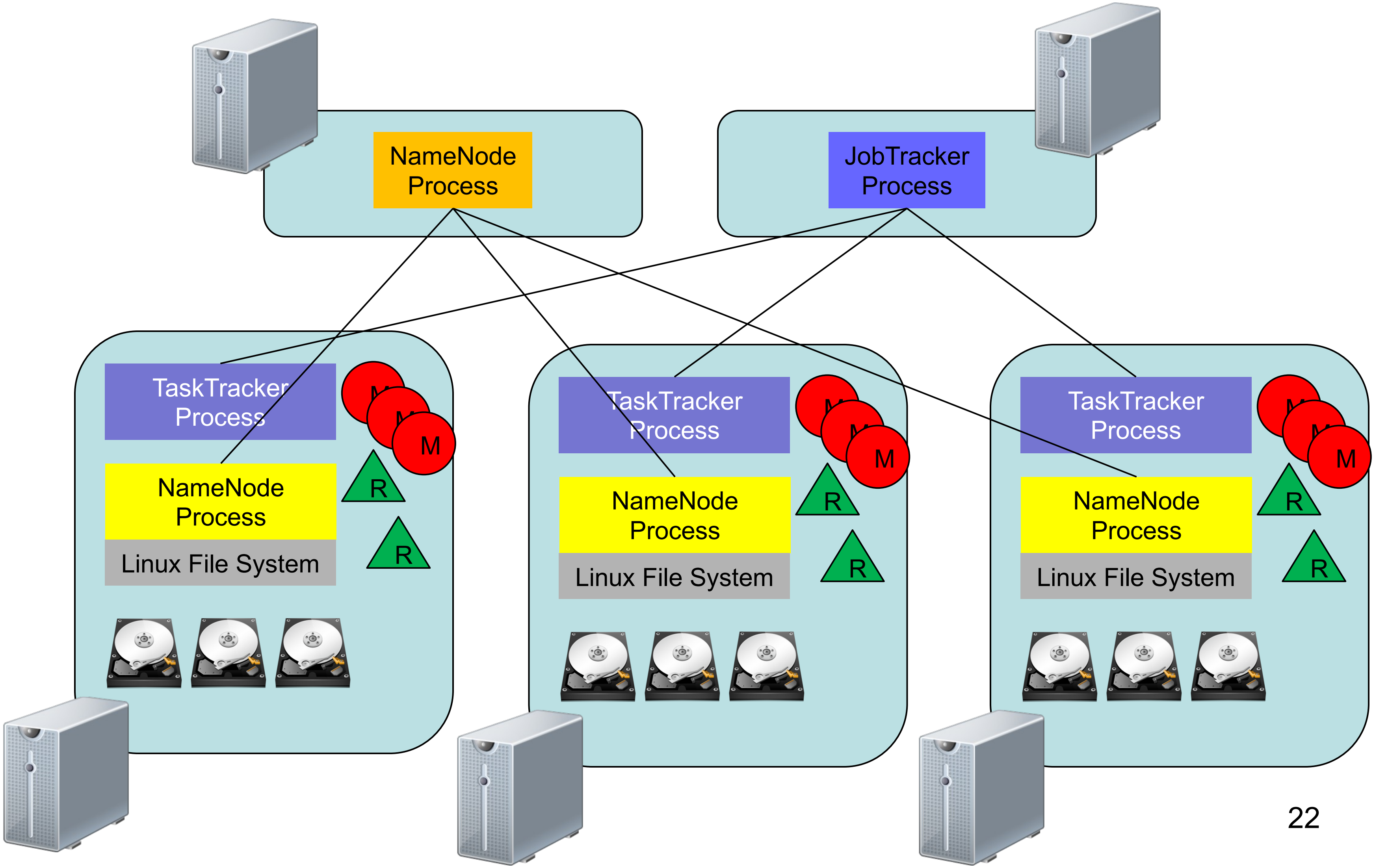
Job Tracker

- **One Job Tracker per Hadoop instance**
- Manage Job life cycle
- Receive and accepts job submissions from clients
- Manages resources and scheduling for each task
- Communicates with Task Trackers to deploy and run tasks
- Attempts to assign tasks to support Data Locality.

Task Tracker

- **One Task Tracker per node**
- Runs and manages individual tasks (maps/reduces) locally
- Communicates progress of tasks back to Job Tracker

HDFS + MapReduce



Apache Hadoop

Distributed Storage
HDFS

Distributed Computation
MapReduce

MapReduce Implementation

- Programmer provider the Map and Reduce functions.
- Everything else is handled by Hadoop M/R
  - Let’s look into details!

HADOOP: The Definitive Guide (4th edition 2016) by Tom White.

MapReduce Implementation

Lets think about it…

- The maps need to finish all, before the reducers can proceed (Barrier)
  - Intermediate results are copied earlier
- Keys arrive at each reducer in sorted order
  - No enforced ordering across reducers

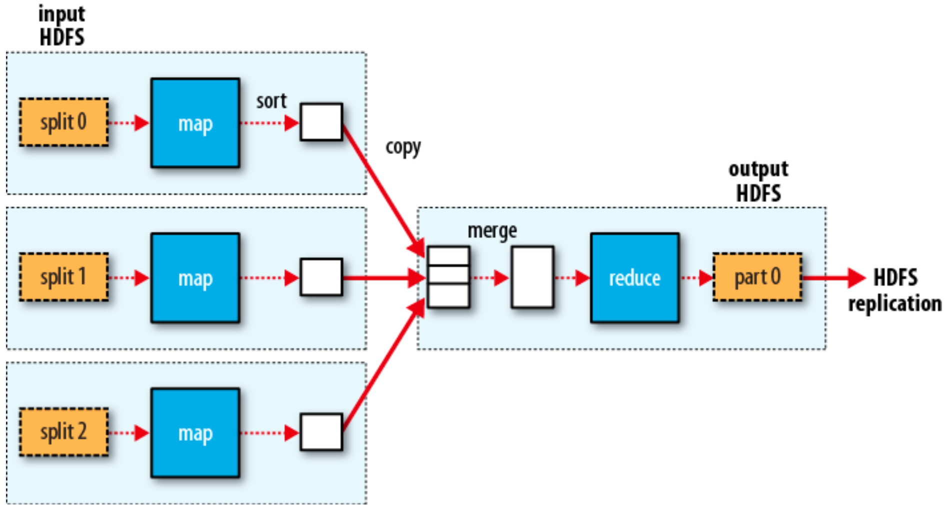
MapReduce Implementation

- Input reader
  - Divide input into splits, assign each split to a Map task
- Map task
  - Apply the Map function to each split
  - Each Map function returns a list of (key, value) pairs
- Shuffle
  - Sort groups the same keys together, and prepares for aggregation
  - Shuffle distributes sorting and aggregation to many reducers
  - All records for key *k* are directed to the same reduce processor
- Reduce task
  - Apply the Reduce function to each key
  - The result of the Reduce function is a list of (key, value) pairs

MapReduce Implementation

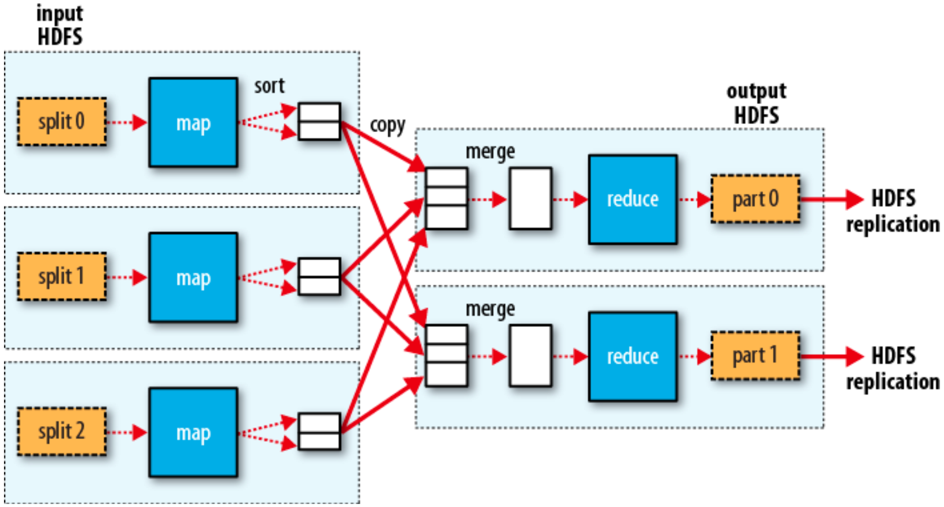
- Handles scheduling
  - Assigns nodes to map and reduce tasks !
- Handles "data distribution"
  - Moves processes to data
- Handles synchronization
  - Gathers, sorts, and shuffles intermediate data !
- Handles errors and faults
- Detects worker failures and restarts
- Everything happens on top of HDFS
  - Intermediary results, shuffling partitions etc

The Anatomy of a M/R Execution
Single reducer



From Tom White 2013

The Anatomy of a M/R Execution multiple reducers



From Tom White 2013

Shuffle and Sort
On the map instance

- Buffering
  - Map outputs are buffered in memory in a **circular buffer** (100 MB)
  - When buffer reaches 80% capacity, contents are “spilled” to disk

- Hadoop will merge all the spill files on a given datanode into a single file
  - This single file is both sorted and partitioned based on number of reducers.
  - in-memory sort the chunk of the buffer (quicksort)

- If the user specifies a combiner then the SPILLING thread, before writing the tuples to disk, executes the combiner on the tuples contained in each partition.

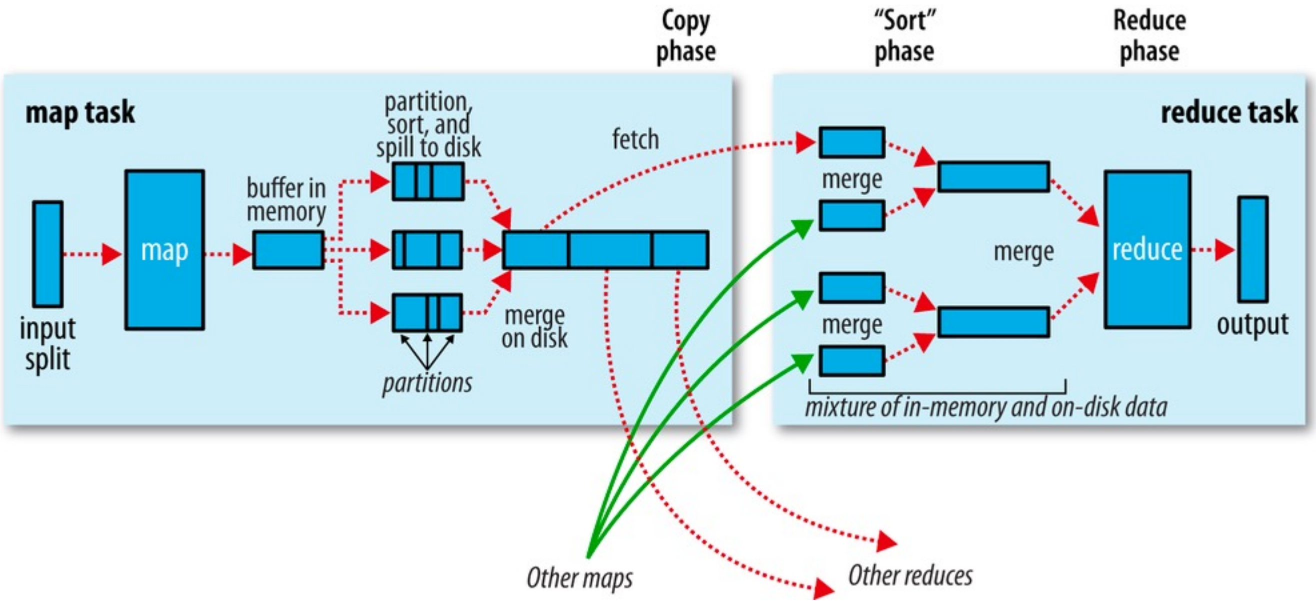
Combiner? MapReduce Programming Optimization

- **combiner** (key, value) → (key, value)
  - A local reducers that run in memory after the map phase
  - Usually the same function as reducer
  - When applicable, it reduces network traffic

Shuffle and Sort On the Reduce instance

- First, map outputs are copied over to reducer machine
- "Sort" is a multi-pass merge of map outputs (happens in memory and on disk)
- **Final merge pass goes directly into reducer**

The Anatomy of a M/R Execution



From Tom White 2016

How to set the number of Mappers and Reducers?

- The number of mappers usually depends on the size of the input data
  - Usually 1 map per block.
  - E.g., For a 1TB input file we can have 7’813 Mappers
    - when block size is 128MB
- Reducers will depend on the expected results
  - 1 Reducer
    - means 1 machine
    - All the key/values will go to the same reducer, bottleneck
  - Too many reducers will create an extensive network traffic
  - Strategy: set depending on the expected number of keys