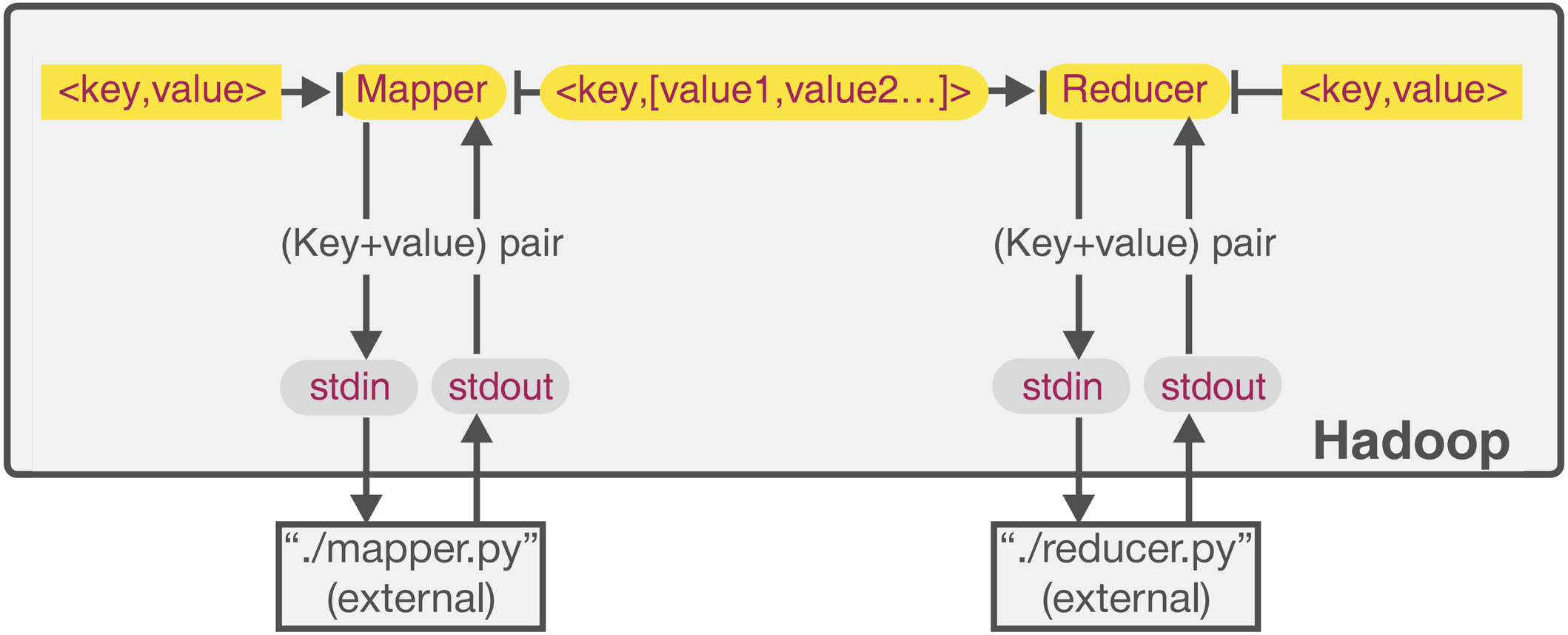
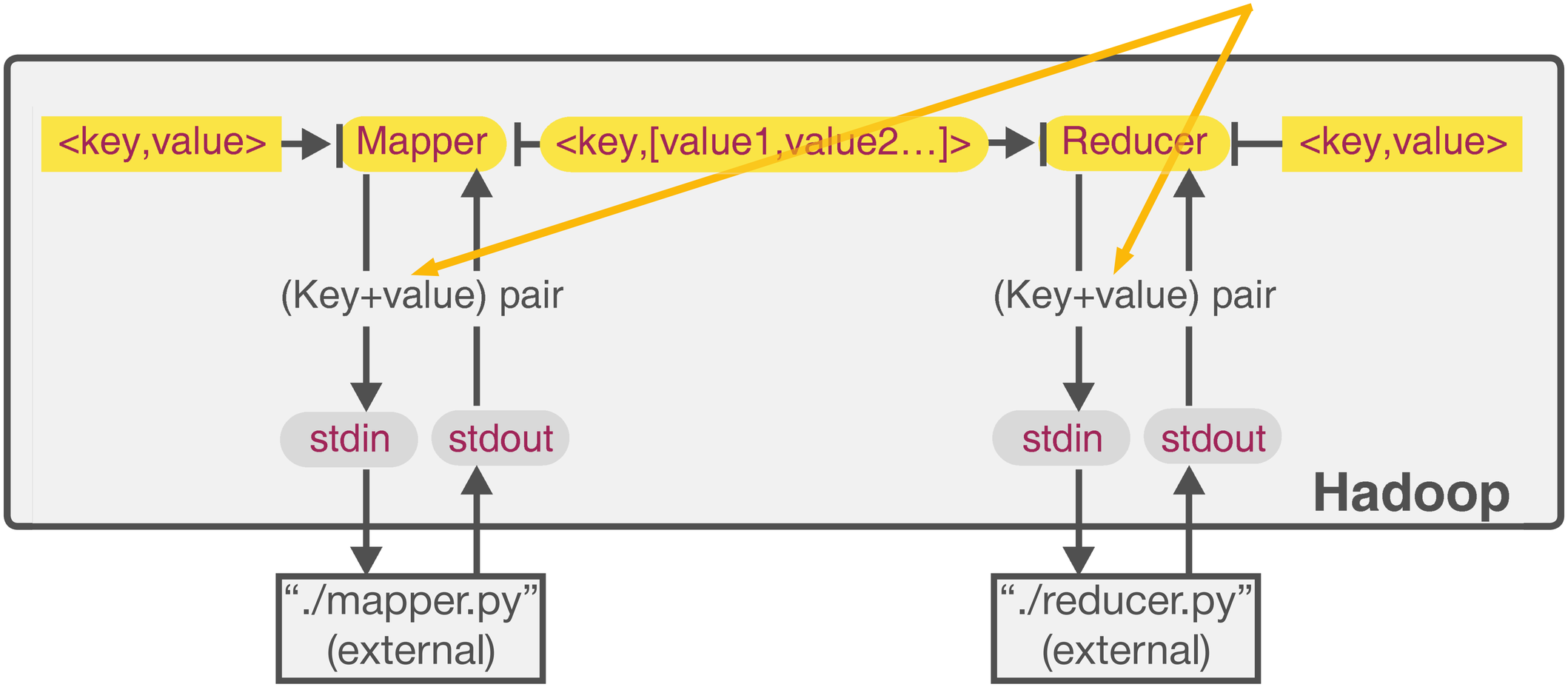
# MapReduce

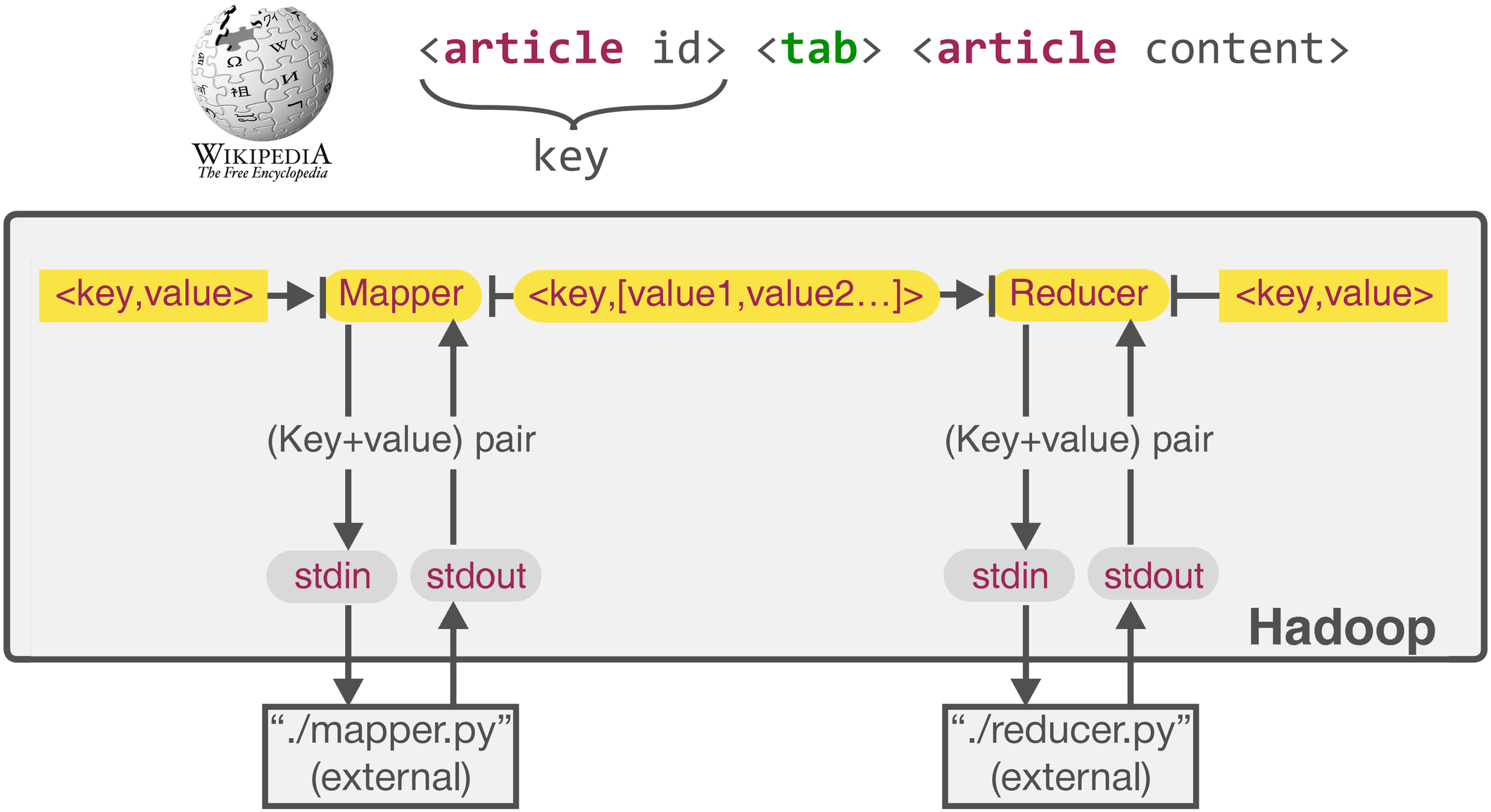
## WordCount in Python

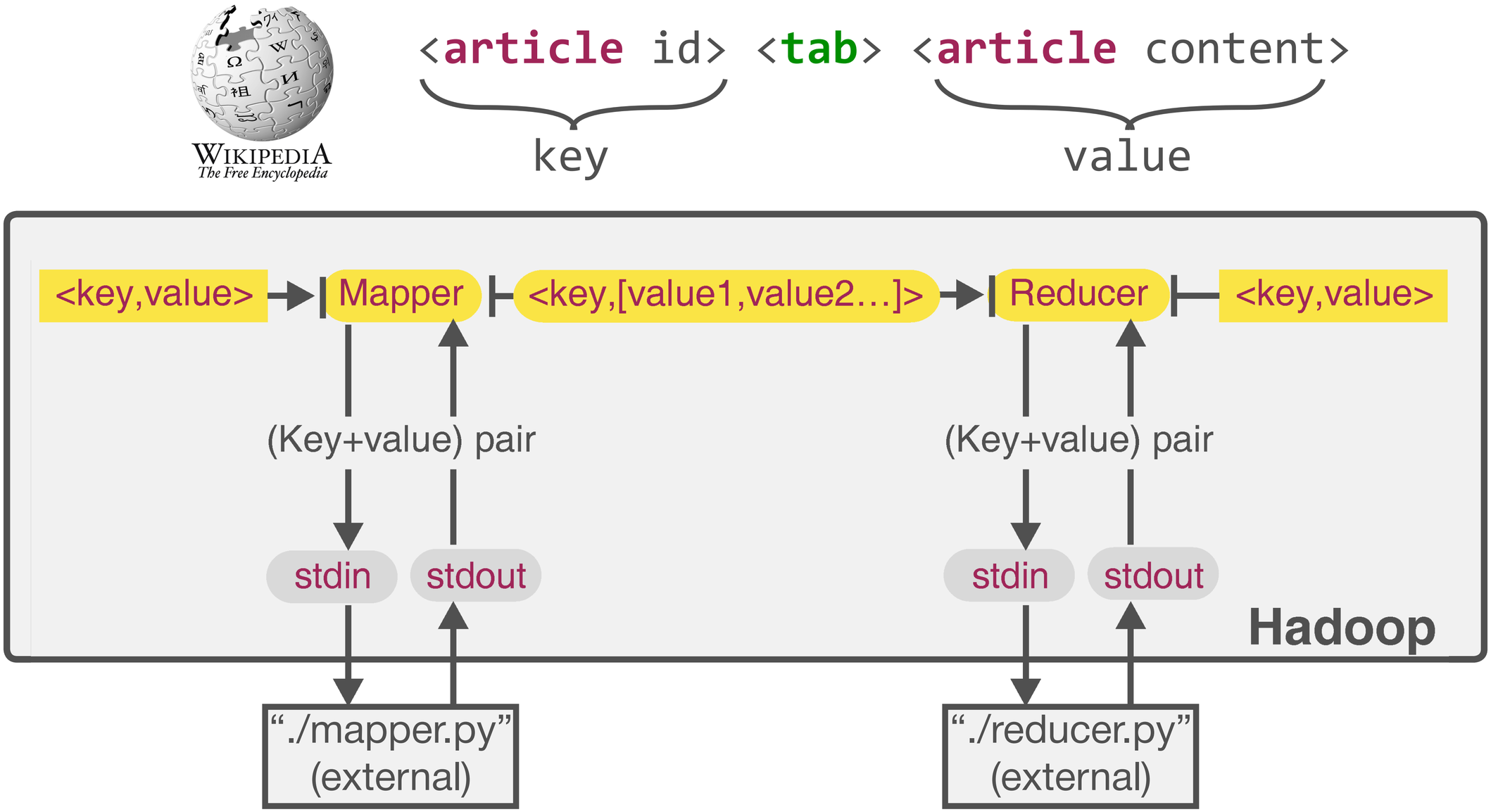
# WordCount

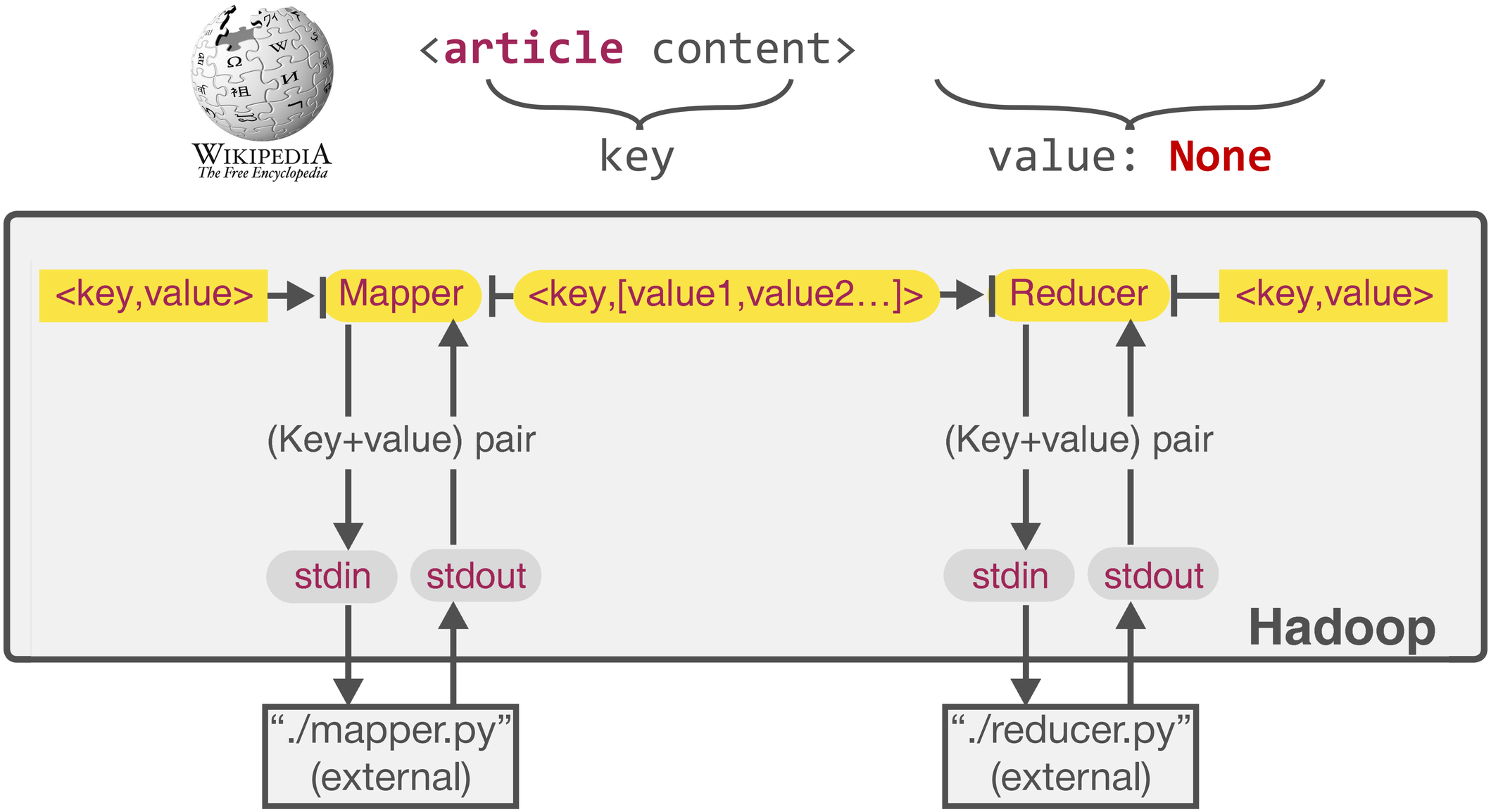

<key,value>
Mapper
<key,[value1,value2…]>
Reducer
<key,value>
(Key+value) pair
(Key+value) pair
stdin
stdout
stdin
stdout
Hadoop
"./mapper.py"
(external)
"./reducer.py"
(external)

# WordCount


<key,value>
Mapper
<key,[value1,value2…]>
Reducer
<key,value>
(Key+value) pair
(Key+value) pair
stdin
stdout
stdin
stdout
Hadoop
"./mapper.py"
(external)
"./reducer.py"
(external)


WIKIPEDIA
The Free Encyclopedia
<article id> <tab> <article content>
key
<key,value>
Mapper
<key,[value1,value2…]>
Reducer
<key,value>
(Key+value) pair
(Key+value) pair
stdin
stdout
stdin
stdout
Hadoop
"./mapper.py"
(external)
"./reducer.py"
(external)


WIKIPEDIA
The Free Encyclopedia
<article id> <tab> <article content>
key
value
<key,value>
Mapper
<key,[value1,value2…]>
Reducer
<key,value>
(Key+value) pair
(Key+value) pair
stdin
stdout
stdin
stdout
Hadoop
"./mapper.py"
(external)
"./reducer.py"
(external)


WIKIPEDIA
The Free Encyclopedia
<article content>
key
value: None
<key,value>
Mapper
<key,[value1,value2…]>
Reducer
<key,value>
(Key+value) pair
(Key+value) pair
stdin
stdout
stdin
stdout
Hadoop
"./mapper.py"
(external)
"./reducer.py"
(external)

WIKIPEDIA
The Free Encyclopedia

$$\underbrace{\langle \textbf{article}\ \text{id} \rangle}_{\text{key}}\ \langle \textbf{tab} \rangle\ \underbrace{\langle \textbf{article}\ \text{content} \rangle}_{\text{value}}$$

```
from __future__ import print_function
import sys

for line in sys.stdin:
  article_id, content = line.split("\t", 1)
    words = content.split()
    for word in words:
        print(word, 1, sep="\t")
```

WIKIPEDIA
The Free Encyclopedia

$$\underbrace{\langle\text{article id}\rangle}_{\text{key}}\ \langle\text{tab}\rangle\ \underbrace{\langle\text{article content}\rangle}_{\text{value}}$$

```
from __future__ import print_function
import sys

for line in sys.stdin:
  article_id, content = line.split("\t", 1)
    words = content.split()
     for word in words:
         print(word, 1, sep="\t")
```

WIKIPEDIA
The Free Encyclopedia

$$\underbrace{\langle\text{article id}\rangle}_{\text{key}}\ \langle\text{tab}\rangle\ \underbrace{\langle\text{article content}\rangle}_{\text{value}}$$

```
from __future__ import print_function
import sys

for line in sys.stdin:
  article_id, content = line.split("\t", 1)
    words = content.split()
    for word in words:
        print(word, 1, sep="\t")
```

```
yarn jar $HADOOP_STREAMING_JAR \
          -files mapper.py \
          -mapper 'python mapper.py' \
          -numReduceTasks 0 \
          -input /data/wiki/en_articles \
          -output word_count
```

```
yarn jar $HADOOP_STREAMING_JAR \
          -files mapper.py \
          -mapper 'python mapper.py' \
          -numReduceTasks 0 \
          -input /data/wiki/en_articles \
          -output word_count
```

```
$ hdfs dfs -text /data/wiki/en_articles/* | head -c 80
12 <tab> Anarchism        Anarchism is often defined as a political
philosophy which …
```

```
yarn jar $HADOOP_STREAMING_JAR \
          -files mapper.py \
          -mapper 'python mapper.py' \
          -numReduceTasks 0 \
          -input /data/wiki/en_articles \
          -output word_count
```

```
$ hdfs dfs -text /data/wiki/en_articles/* | head -c 80
12 <tab> Anarchism        Anarchism is often defined as a political
philosophy which …
```

```
$ hdfs dfs -ls -h word_count
Found 3 items
-rw-r--r--  3 adral adral  0 2017-03-22 11:40 word_count/_SUCCESS
-rw-r--r--  3 adral adral  47.8 M 2017-03-22 11:40 word_count/part-00000
-rw-r--r--  3 adral adral  47.9 M 2017-03-22 11:40 word_count/part-00001
```

```
yarn jar $HADOOP_STREAMING_JAR \
          -files mapper.py \
          -mapper 'python mapper.py' \
          -numReduceTasks 0 \
          -input /data/wiki/en_articles \
          -output word_count
```

```
$ hdfs dfs -text /data/wiki/en_articles/*
| head -c 80
12 <tab> Anarchism          Anarchism is
often defined as a political philosophy
which …
```

```
$ hdfs dfs -text
    word_count/part-… | head -5
```

| part-00000 | part-00001 |
|---|---|
| Basel 1 | Anarchism 1 |
| Basel 1 | Anarchism 1 |
| ( 1 | is 1 |
| ) 1 | often 1 |
| or 1 | defined 1 |
| … | … |

```
yarn jar $HADOOP_STREAMING_JAR \
          -files mapper.py \
          -mapper 'python mapper.py' \
          -numReduceTasks 0 \
          -input /data/wiki/en_articles \
          -output word_count
```

```
$ hdfs dfs -text /data/wiki/en_articles/*
| head -c 80
12 <tab> Anarchism          Anarchism is
often defined as a political philosophy
which …
```

```
$ hdfs dfs -text
    word_count/part-… | head -5
```

| part-00000 | part-00001 |
|---|---|
| Basel 1 | Anarchism 1 |
| Basel 1 | Anarchism 1 |
| ( 1 | is 1 |
| ) 1 | often 1 |
| or 1 | defined 1 |
| … | … |

```
yarn jar $HADOOP_STREAMING_JAR \
          -files mapper.py \
          -mapper 'python mapper.py' \
          -numReduceTasks 1 \
          -input /data/wiki/en_articles \
          -output word_count
```

```
yarn jar $HADOOP_STREAMING_JAR \
          -files mapper.py \
          -mapper 'python mapper.py' \
          -numReduceTasks 1 \
          -input /data/wiki/en_articles \
          -output word_count
```

```
$ hdfs dfs -text word_count/part-00000 | head
! 1
! 1
! 1
! 1
! 1
…
```

WIKIPEDIA
The Free Encyclopedia

$$\underbrace{\langle \text{article id} \rangle}_{\text{key}} \ \langle \text{tab} \rangle \ \underbrace{\langle \text{article content} \rangle}_{\text{value}}$$

```python
from __future__ import print_function
import re
import sys

for line in sys.stdin:
 article_id, content = line.split("\t", 1)
    words = re.split("\W+", content)
    for word in words:
        if word:
           print(word, 1, sep="\t")
```


<article id> <tab> <article content>

key value

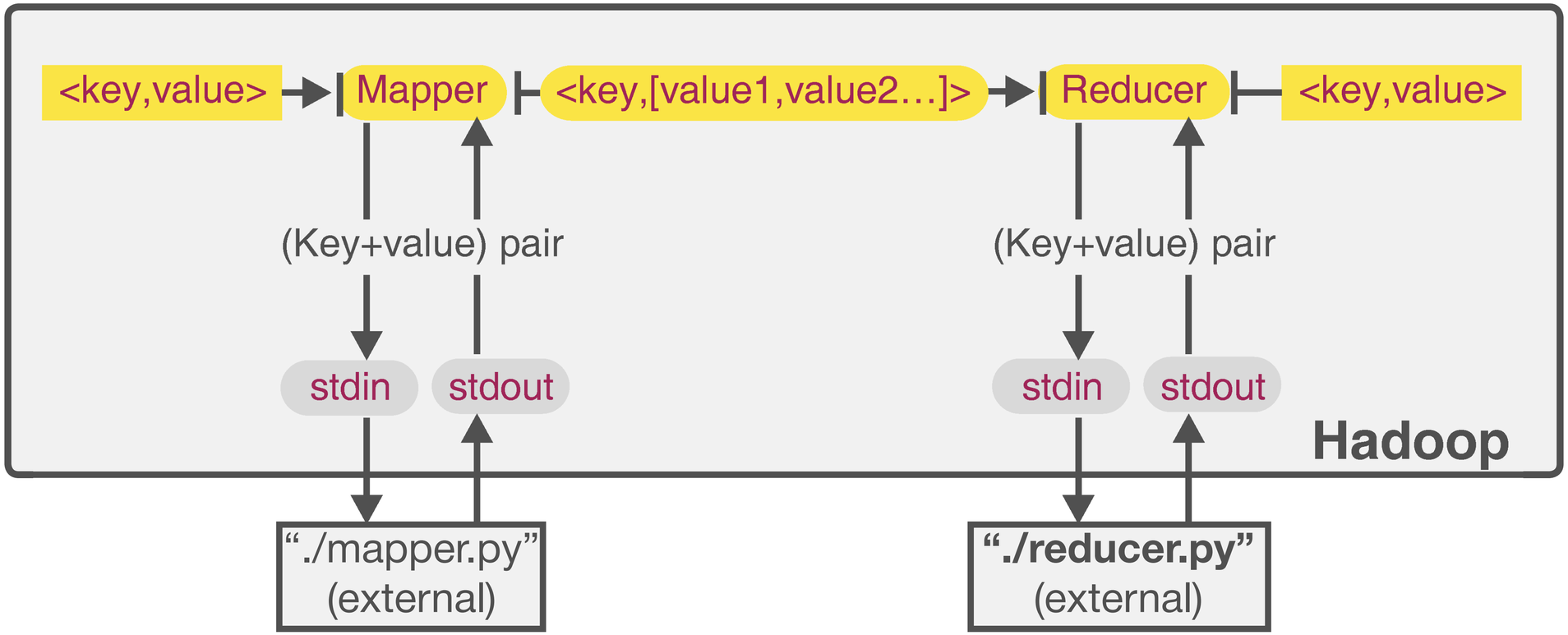
```
from __future__ import print_function
import re
import sys

for line in sys.stdin:
 article_id, content = line.split("\t", 1)
    words = re.split("\W+", content)
    for word in words:
        if word:
           print(word, 1, sep="\t")
```

```
yarn jar $HADOOP_STREAMING_JAR \
            -files mapper.py \
            -mapper 'python mapper.py' \
            -numReduceTasks 1 \
            -input /data/wiki/en_articles \
            -output word_count
```

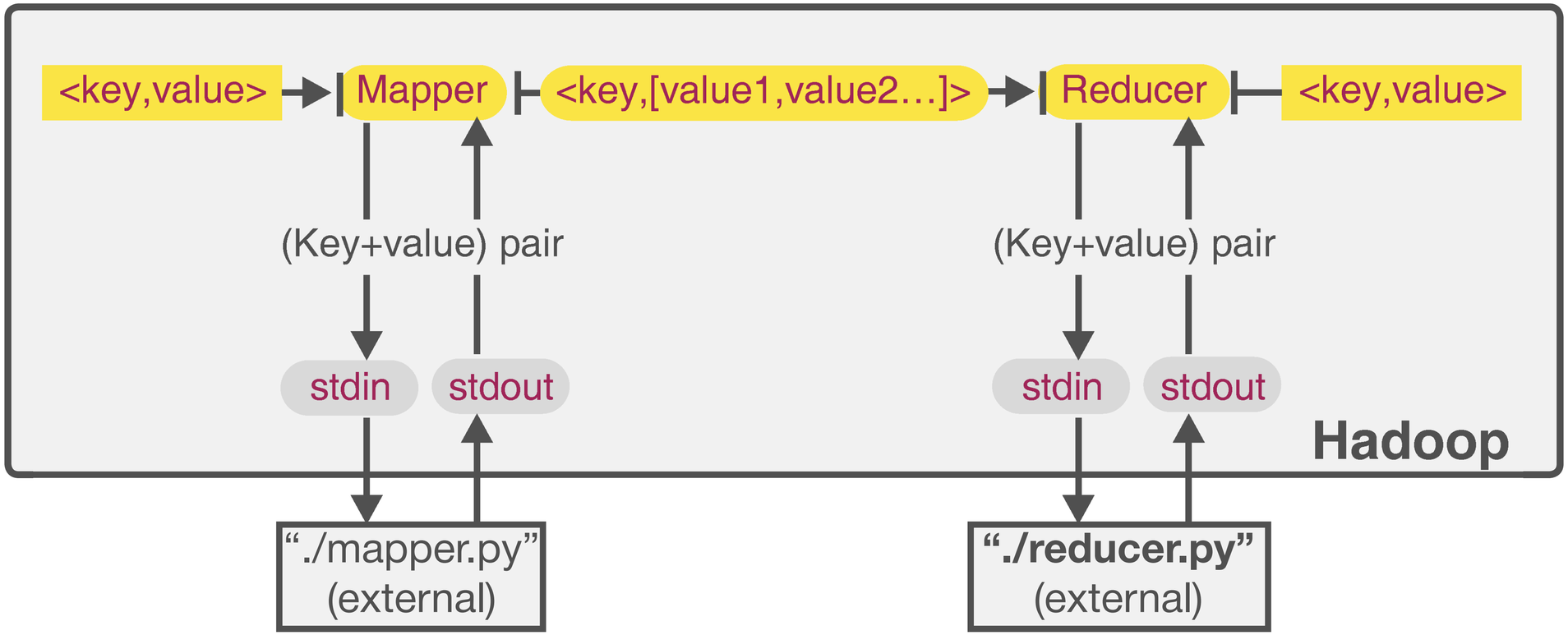
```
$ hdfs dfs -text word_count/part-00000 | head -4
0  1
0  1
0  1
0  1
```

```
$ hdfs dfs -tail word_count/part-00000 | tail -4
zyu1 1
zyu1 1
zz  1
zz  1
```

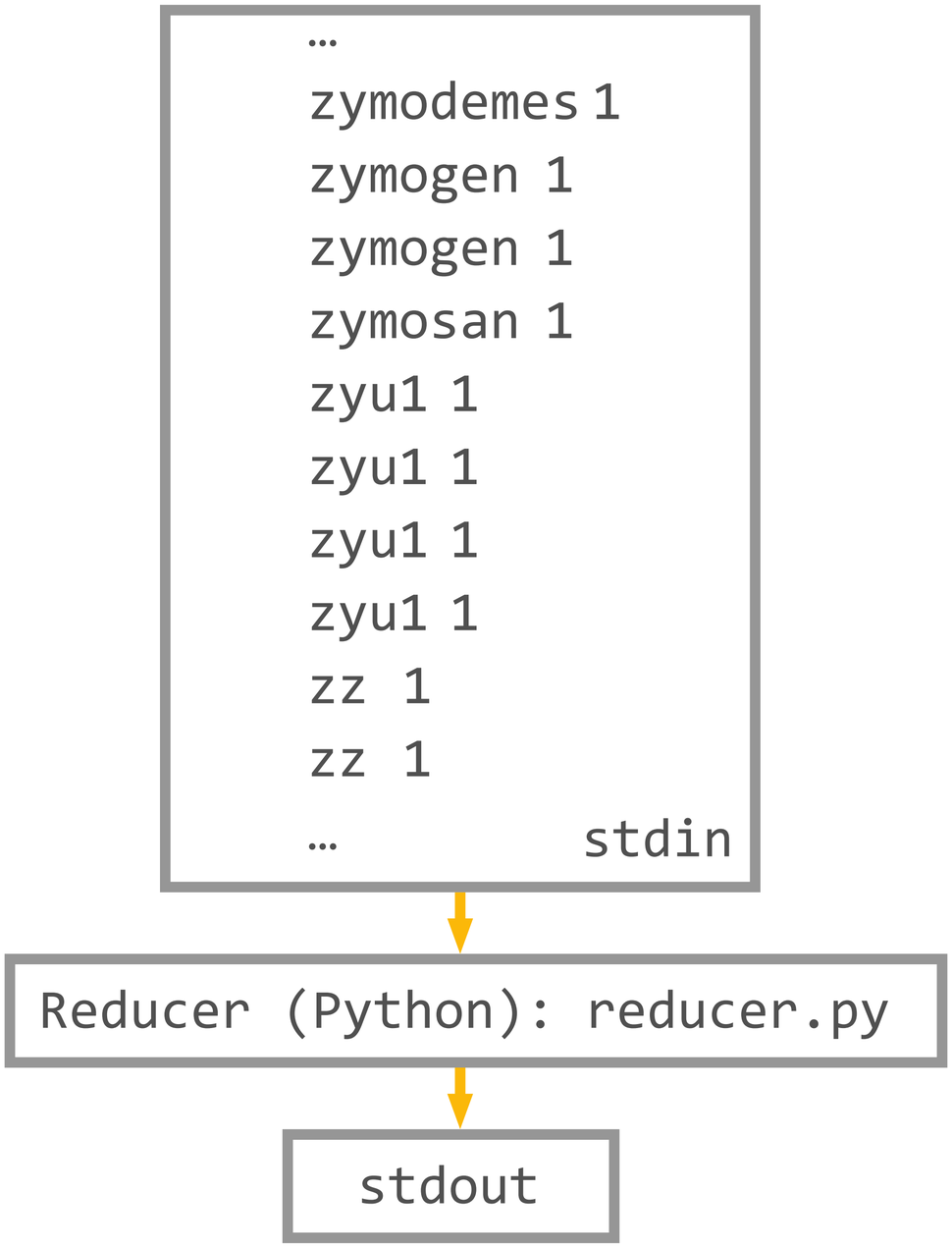
# WordCount

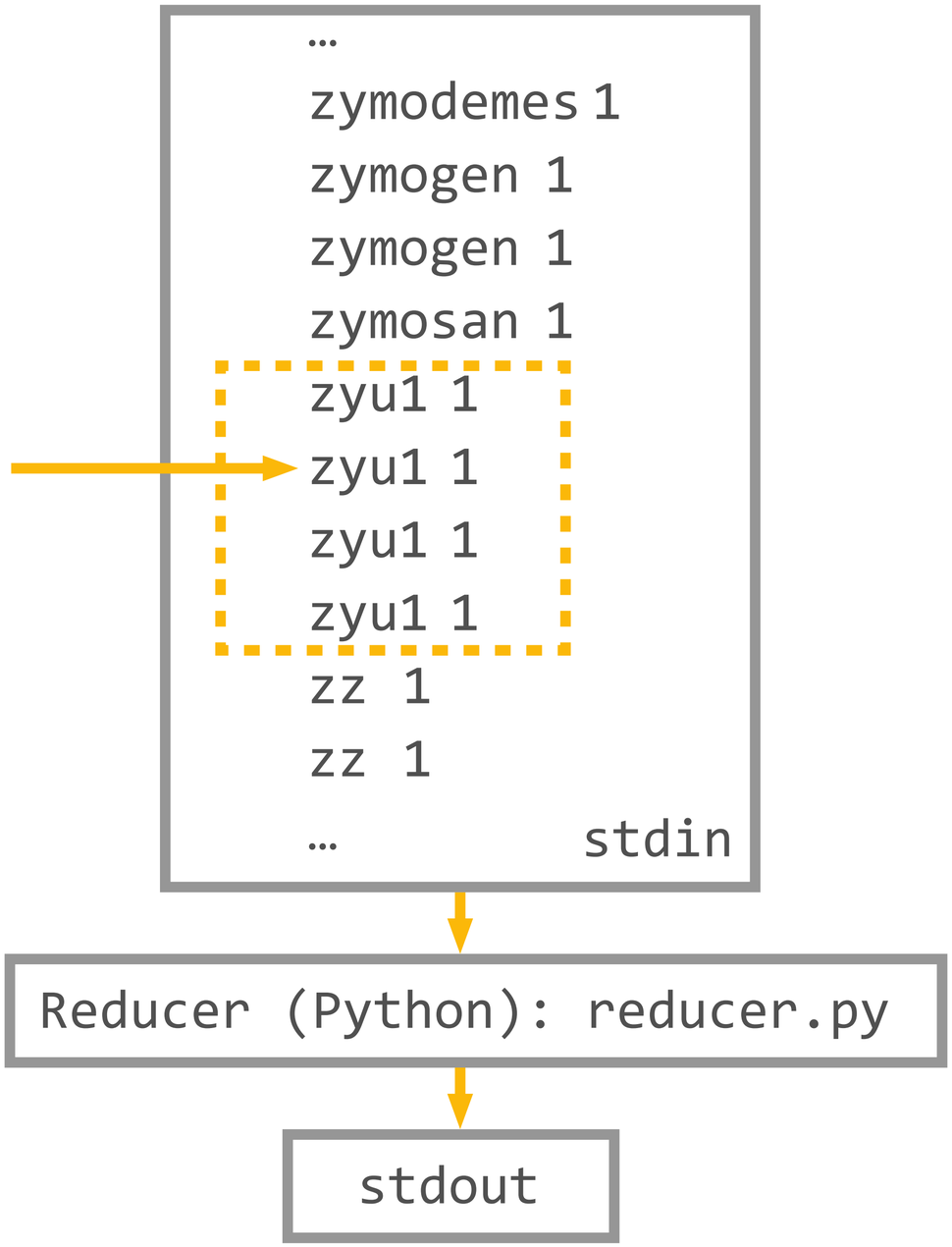

<key,value>
Mapper
<key,[value1,value2…]>
Reducer
<key,value>
(Key+value) pair
(Key+value) pair
stdin
stdout
stdin
stdout
Hadoop
"./mapper.py"
(external)
"./reducer.py"
(external)

# WordCount




- define input format
- **aggregate sorted data by key**
- process data
- define output format


…
zymodemes 1
zymogen 1
zymogen 1
zymosan 1
zyu1 1
zyu1 1
zyu1 1
zyu1 1
zz 1
zz 1
…
stdin
Reducer (Python): reducer.py
stdout


…
zymodemes 1
zymogen 1
zymogen 1
zymosan 1
zyu1 1
zyu1 1
zyu1 1
zyu1 1
zz 1
zz 1
…
stdin
Reducer (Python): reducer.py
stdout

```python
from __future__ import print_function
import sys

current_word = None
word_count = 0

for line in sys.stdin:
    word, counts = line.split("\t", 1)
    counts = int(counts)
    if word == current_word:
        word_count += counts
    else:
        if current_word:
            print(current_word, word_count, sep="\t")
        current_word = word
        word_count = counts

if current_word:
    print(current_word, word_count, sep="\t")
```

```python
from __future__ import print_function
import sys

current_word = None
word_count = 0

for line in sys.stdin:
    word, counts = line.split("\t", 1)
    counts = int(counts)
    if word == current_word:
        word_count += counts
    else:
        if current_word:
            print(current_word, word_count, sep="\t")
        current_word = word
        word_count = counts

if current_word:
    print(current_word, word_count, sep="\t")
```

```python
from __future__ import print_function
import sys

current_word = None
word_count = 0

for line in sys.stdin:
    word, counts = line.split("\t", 1)
    counts = int(counts)
    if word == current_word:
        word_count += counts
    else:
        if current_word:
            print(current_word, word_count, sep="\t")
        current_word = word
        word_count = counts

if current_word:
    print(current_word, word_count, sep="\t")
```

```python
from __future__ import print_function
import sys

current_word = None
word_count = 0

for line in sys.stdin:
    word, counts = line.split("\t", 1)
    counts = int(counts)
    if word == current_word:
        word_count += counts
    else:
        if current_word:
            print(current_word, word_count, sep="\t")
        current_word = word
        word_count = counts

if current_word:
    print(current_word, word_count, sep="\t")
```

```
…
zymosan 1
zyu1 1
zyu1 1
zyu1 1
zyu1 1
zz 1
…
```

```python
from __future__ import print_function
import sys

current_word = None
word_count = 0

for line in sys.stdin:
    word, counts = line.split("\t", 1)
    counts = int(counts)
    if word == current_word:
        word_count += counts
    else:
        if current_word:
            print(current_word, word_count, sep="\t")
        current_word = word
        word_count = counts

if current_word:
    print(current_word, word_count, sep="\t")
```

```
…
zymosan 1
zyu1 1
zyu1 1
zyu1 1
zyu1 1
zz 1
…
```

```python
from __future__ import print_function
import sys

current_word = None
word_count = 0

for line in sys.stdin:
    word, counts = line.split("\t", 1)
    counts = int(counts)
    if word == current_word:
        word_count += counts
    else:
        if current_word:
            print(current_word, word_count, sep="\t")
        current_word = word
        word_count = counts

if current_word:
    print(current_word, word_count, sep="\t")
```

```
yarn jar $HADOOP_STREAMING_JAR \
            -files mapper.py,reducer.py \
            -mapper 'python mapper.py' \
            -reducer 'python reducer.py' \
            -numReduceTasks 1 \
            -input /data/wiki/en_articles \
            -output word_count
```

```
$ hdfs dfs -ls -h word_count
Found 2 items
-rw-r--r--   3 adral adral          0 2017-03-22 13:05 word_count/_SUCCESS
-rw-r--r--   3 adral adral      3.2 M 2017-03-22 13:05 word_count/part-00000
```

```
yarn jar $HADOOP_STREAMING_JAR \
            -files mapper.py,reducer.py \
            -mapper 'python mapper.py' \
            -reducer 'python reducer.py' \
            -numReduceTasks 1 \
            -input /data/wiki/en_articles \
            -output word_count
```

```
$ hdfs dfs -ls -h word_count
Found 2 items
-rw-r--r--   3 adral adral          0 2017-03-22 13:05 word_count/_SUCCESS
-rw-r--r--   3 adral adral      3.2 M 2017-03-22 13:05 word_count/part-00000
```

```
$ hdfs dfs -ls -h word_count
Found 2 items
-rw-r--r--   3 adral adral          0 2017-03-22 13:05 word_count/_SUCCESS
-rw-r--r--   3 adral adral      3.2 M 2017-03-22 13:05 word_count/part-00000
```

```
$ hdfs dfs -text word_count/part-00000
0 14905
00  844
000 8186
…
zymodemes 1
zymogen  2
zymosan  1
zyu1 4
zz  2
```

```
…
zymosan 1
zyu1 1
zyu1 1
zyu1 1
zyu1 1
zz 1
…
```

```
yarn jar $HADOOP_STREAMING_JAR \
            -files mapper.py,reducer.py \
            -mapper 'python mapper.py' \
            -reducer 'python reducer.py' \
            -input /data/wiki/en_articles \
            -output word_count
```

```
yarn jar $HADOOP_STREAMING_JAR \
            -files mapper.py,reducer.py \
            -mapper 'python mapper.py' \
            -reducer 'python reducer.py' \
            -input /data/wiki/en_articles \
            -output word_count
```

```
$ hdfs dfs -ls -h word_count
Found 11 items
-rw-r--r--   3 adral adral          0 2017-03-22 13:19 word_count/_SUCCESS
-rw-r--r--   3 adral adral    331.0 K 2017-03-22 13:18 word_count/part-00000
-rw-r--r--   3 adral adral    332.1 K 2017-03-22 13:18 word_count/part-00001
-rw-r--r--   3 adral adral    331.7 K 2017-03-22 13:18 word_count/part-00002
-rw-r--r--   3 adral adral    329.8 K 2017-03-22 13:18 word_count/part-00003
-rw-r--r--   3 adral adral    326.1 K 2017-03-22 13:18 word_count/part-00004
-rw-r--r--   3 adral adral    332.2 K 2017-03-22 13:18 word_count/part-00005
-rw-r--r--   3 adral adral    332.3 K 2017-03-22 13:18 word_count/part-00006
```

```
$ hdfs dfs -tail word_count/part-… | tail -5
```

| part-00000 | part-00005 |
|---|---|
| … | … |
| zuang 1 | zsu 1 |
| zucchini 5 | zuchetto 1 |
| zuerst 1 | zure 1 |
| zumase 2 | zuurstof 1 |
| zyu 1 4 | zz 2 |

```
$ hdfs dfs -tail word_count/part-… | tail -5
```

| part-00000 | part-00005 |
|---|---|
| … | … |
| | zsu 1 |
| zuang 1 | |
| zucchini 5 | |
| | zuchetto 1 |
| zuerst 1 | |
| | zure 1 |
| … | … |

```
$ hdfs dfs -tail word_count/part-… | tail -5
```

| part-00000 | part-00005 |
|---|---|
| … | … |
| | zsu 1 |
| zuang 1 | |
| zucchini 5 | |
| | zuchetto 1 |
| zuerst 1 | |
| | zure 1 |
| … | … |

see: TotalOrderPartitioner

# Summary

# Summary

- You know **what** MapReduce Streaming **is** and how it works

# Summary

- You know **what** MapReduce Streaming **is** and how it works
- You know **how to write** MapReduce Bash and Python Streaming applications

# Summary

- You know **what** MapReduce Streaming **is** and how it works
- You know **how to write** MapReduce Bash and Python Streaming applications
- You should be able **to solve** WordCount or similar problems in MapReduce in Python **by yourself**