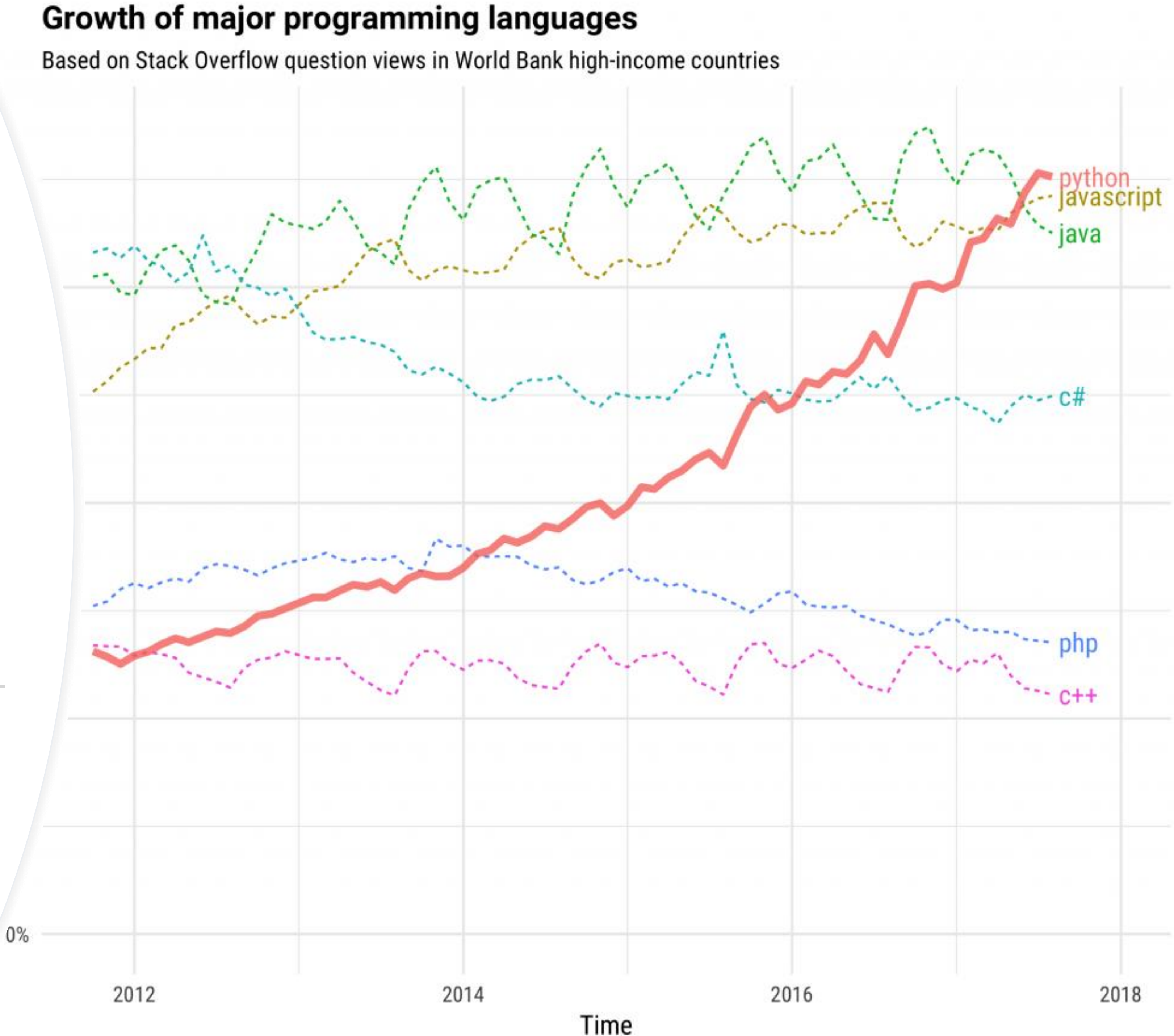
Python已逐漸成為世上最熱門的程式語言

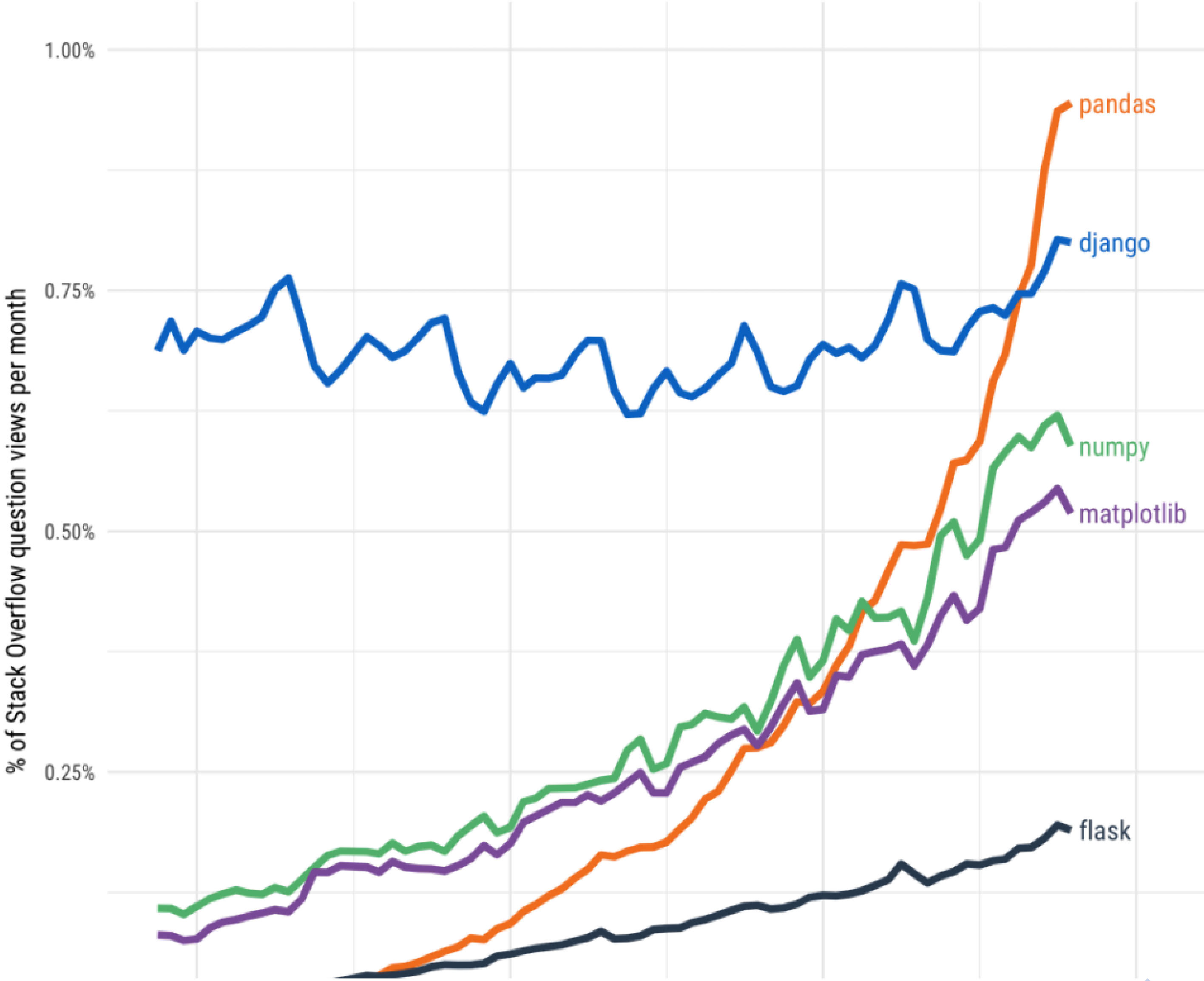


Growth of major programming languages

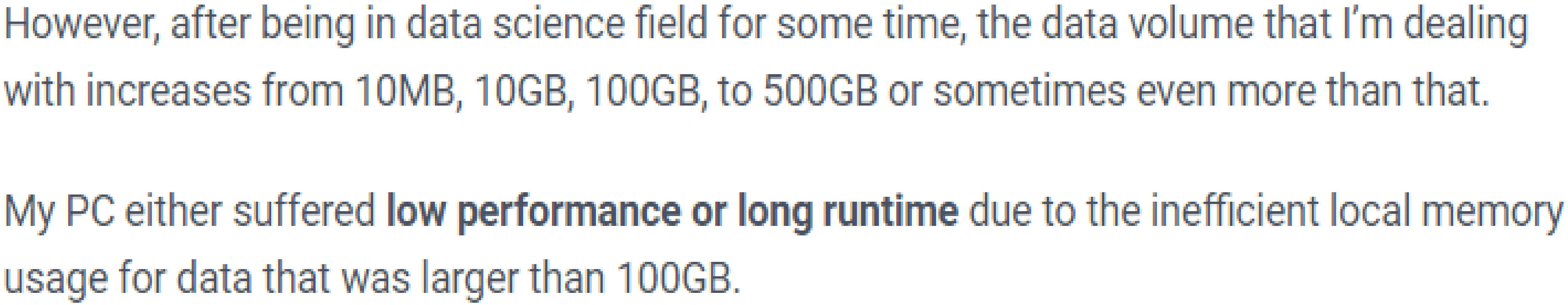
Based on Stack Overflow question views in World Bank high-income countries

Stack Overflow Traffic to Questions About Selected Python Packages

Based on visits to Stack Overflow questions from World Bank high-income countries



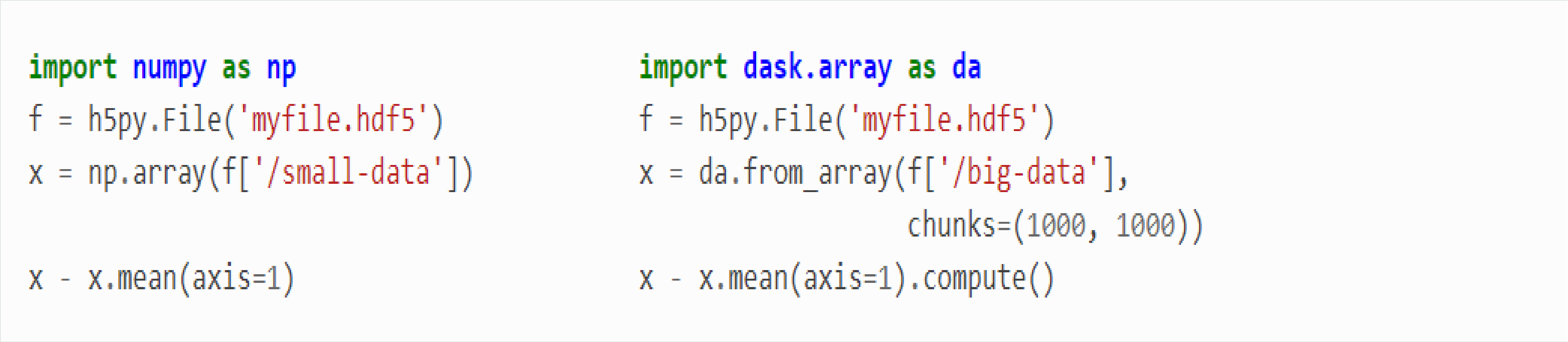
來自網路上的哀號



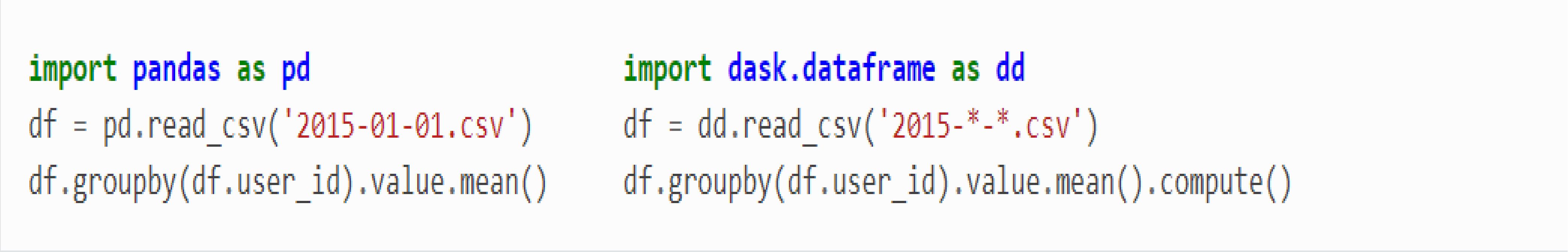
However, after being in data science field for some time, the data volume that I'm dealing with increases from 10MB, 10GB, 100GB, to 500GB or sometimes even more than that.

My PC either suffered **low performance or long runtime** due to the inefficient local memory usage for data that was larger than 100GB.

- 它是一個平行運算的套件
- 使用多核心處理
- 寫法與NumPy, Pandas, Scikit-Learn非常相似

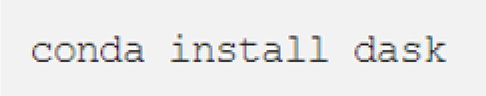


```
import numpy as np                       import dask.array as da
f = h5py.File('myfile.hdf5')             f = h5py.File('myfile.hdf5')
x = np.array(f['/small-data'])           x = da.from_array(f['/big-data'],
                                                           chunks=(1000, 1000))
x - x.mean(axis=1)                       x - x.mean(axis=1).compute()
```



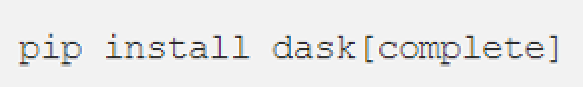
```
import pandas as pd                     import dask.dataframe as dd
df = pd.read_csv('2015-01-01.csv')      df = dd.read_csv('2015-*-*.csv')
df.groupby(df.user_id).value.mean()     df.groupby(df.user_id).value.mean().compute()
```

安裝



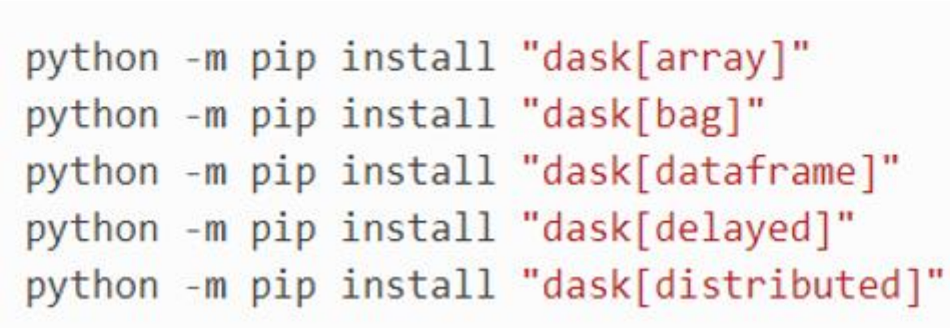
```
conda install dask
```

Or



```
pip install dask[complete]
```

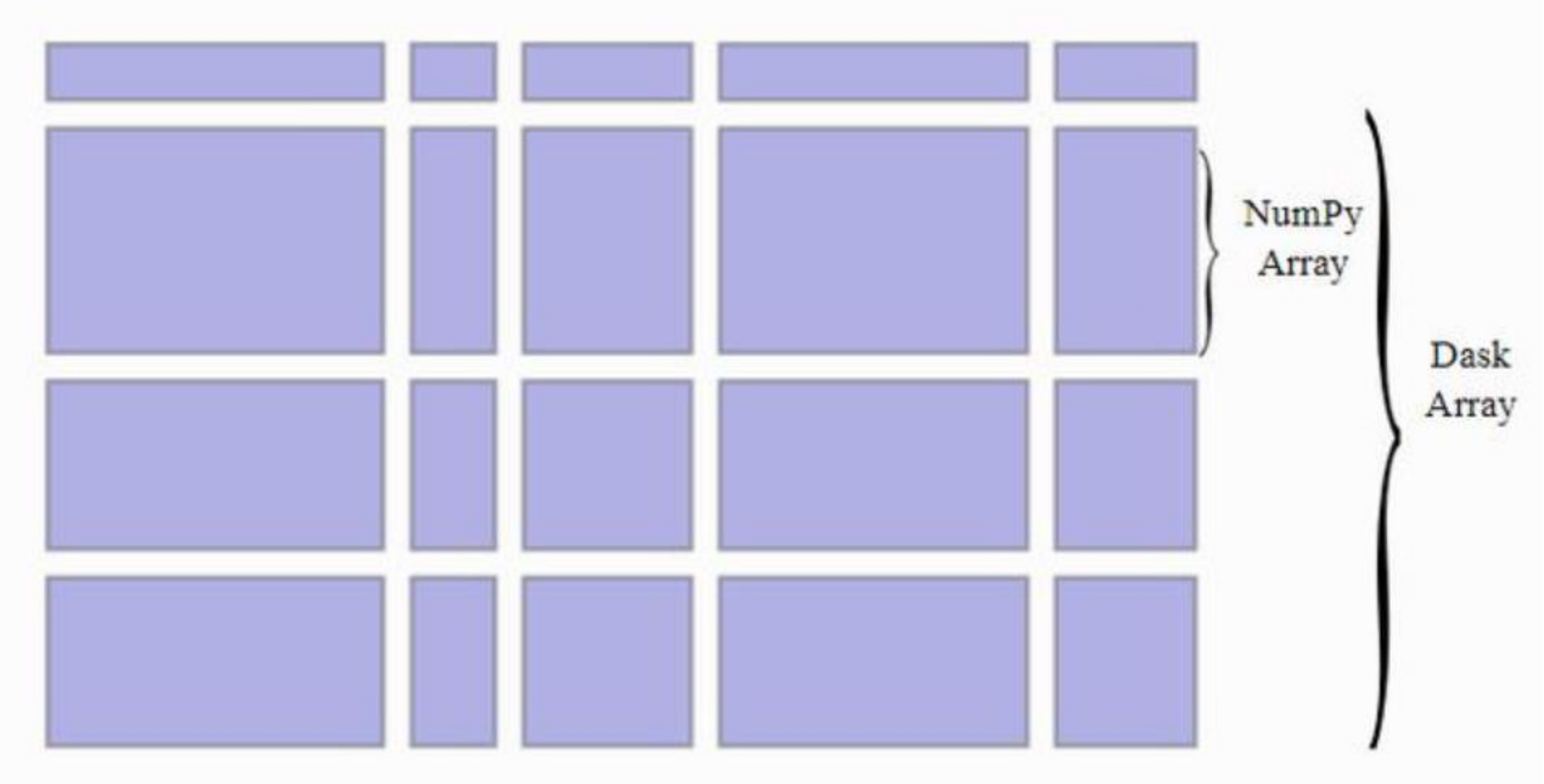
Or

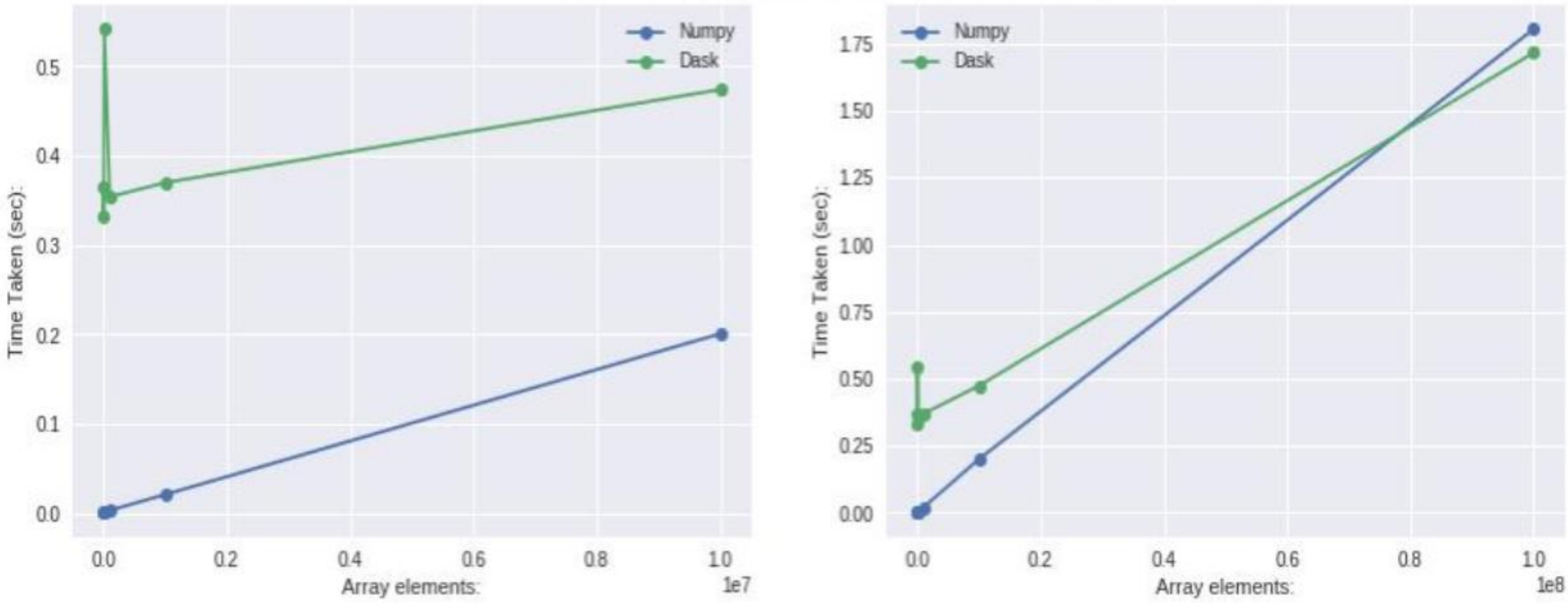


```
python -m pip install "dask[array]"
python -m pip install "dask[bag]"
python -m pip install "dask[dataframe]"
python -m pip install "dask[delayed]"
python -m pip install "dask[distributed]"
```

Dask Arrays

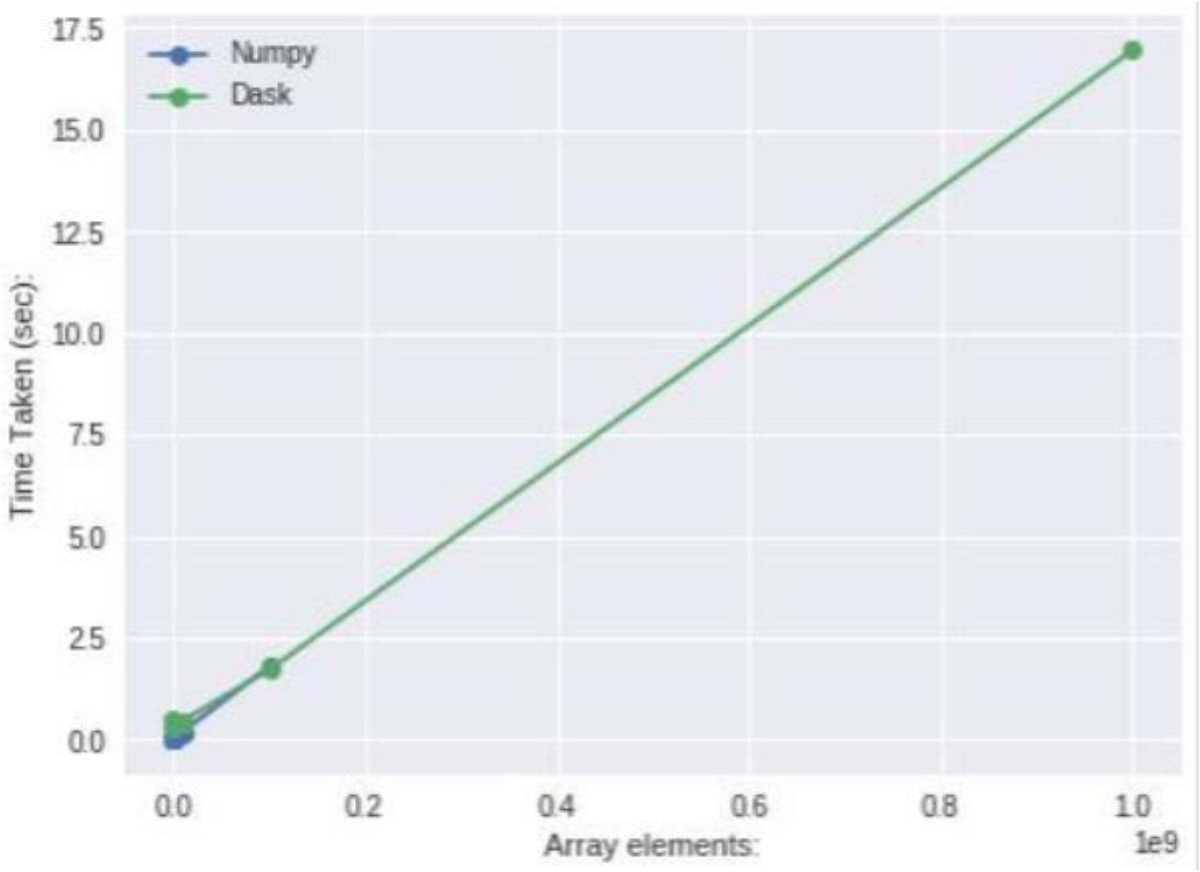
當你的陣列真的很多，且NumPy對此無能為力，Dask將他們分成矩陣塊再平行處理它們。



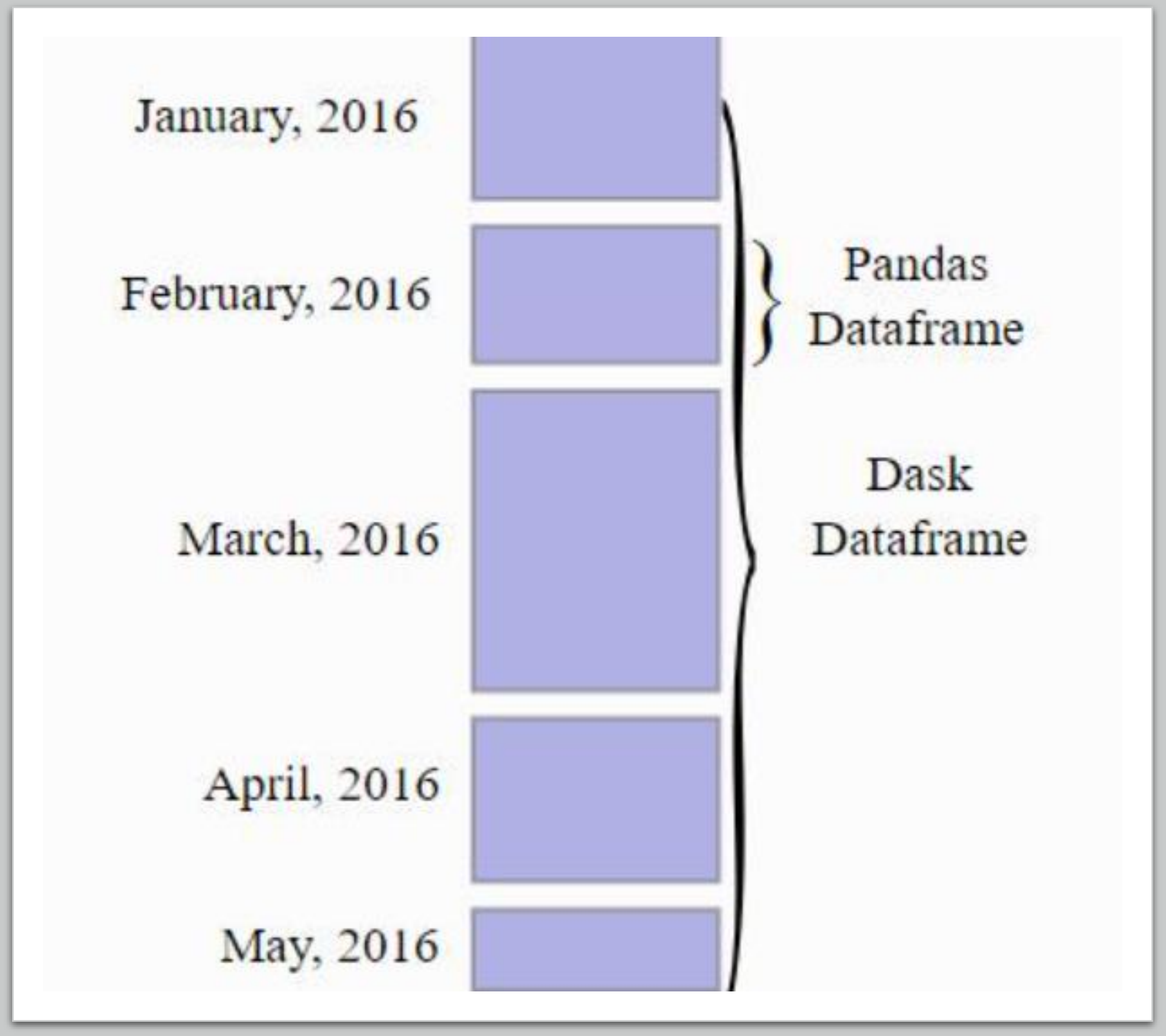


陣列數量小vs陣列數量大

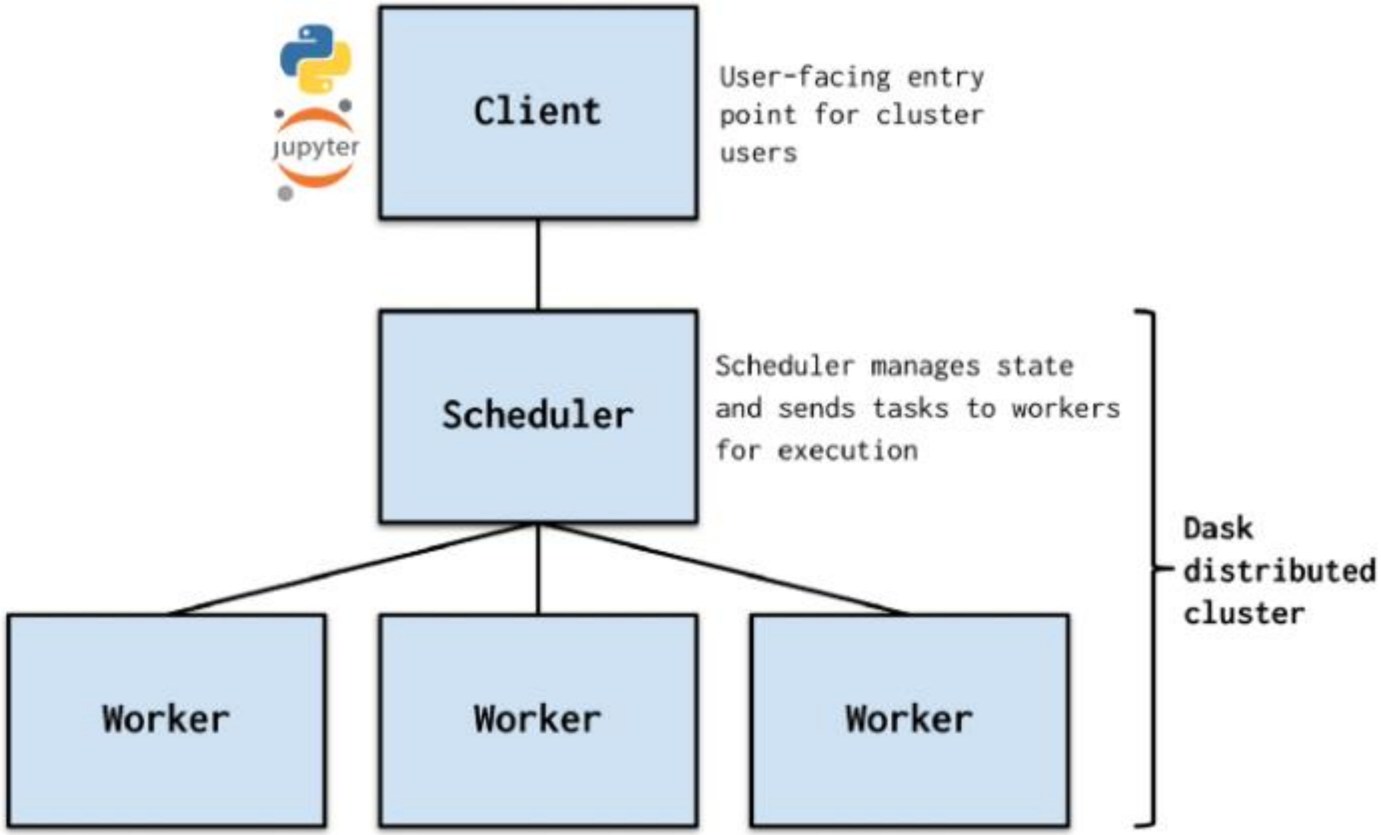
當陣列數量超大


Numpy
Dask
Time Taken (sec):
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1e9
Array elements:

Dask DataFrames



與Dask Array相似，Dask DataFrame將Pandas DataFrame包起來並將這些大塊平行運算。



Client

User-facing entry point for cluster users

Scheduler

Scheduler manages state and sends tasks to workers for execution

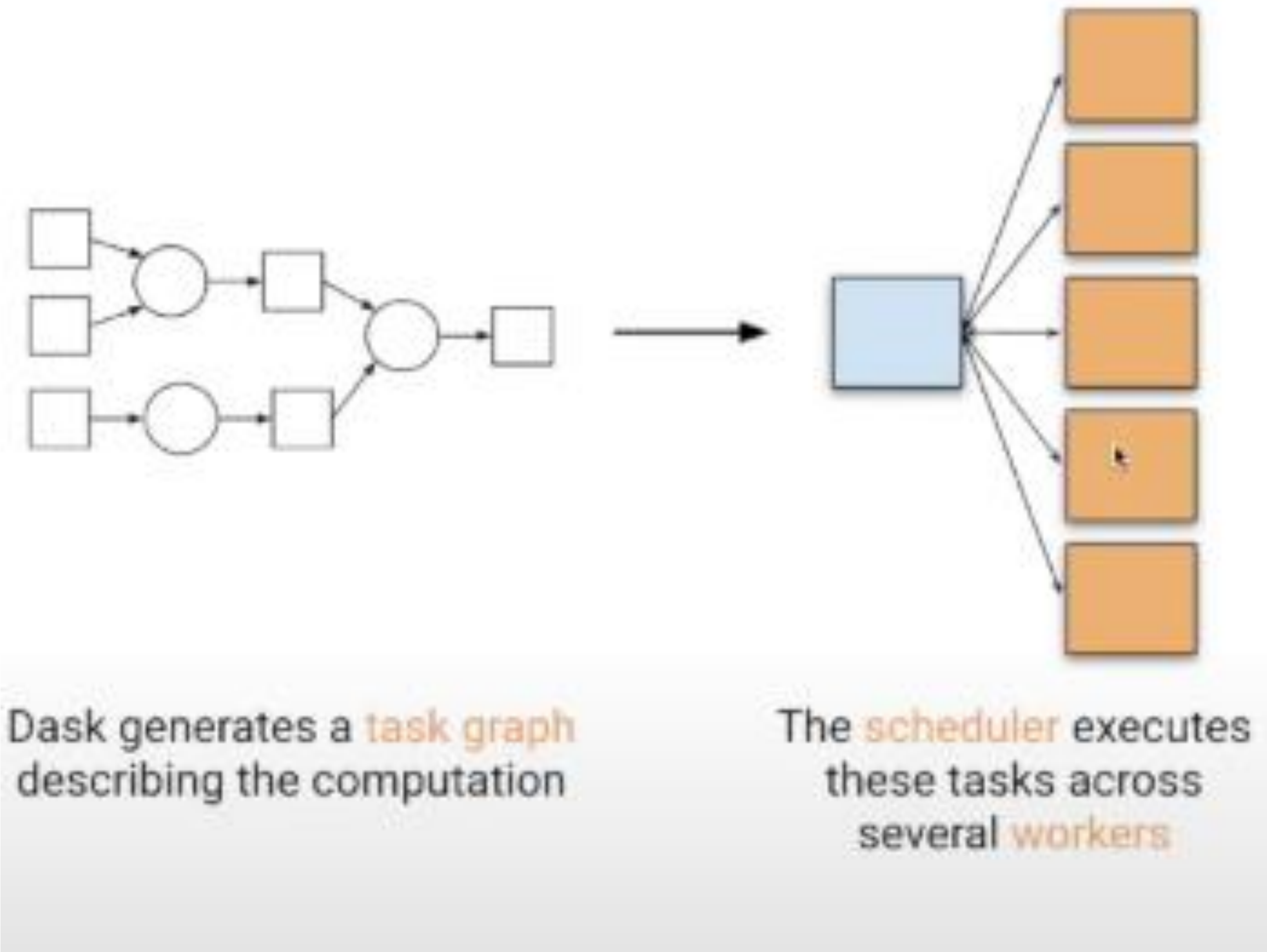
Worker

Worker

Worker

Dask distributed cluster

Workers compute tasks / store and serve computed results to other workers or clients



Dask generates a task graph describing the computation

The scheduler executes these tasks across several workers