MapReduce

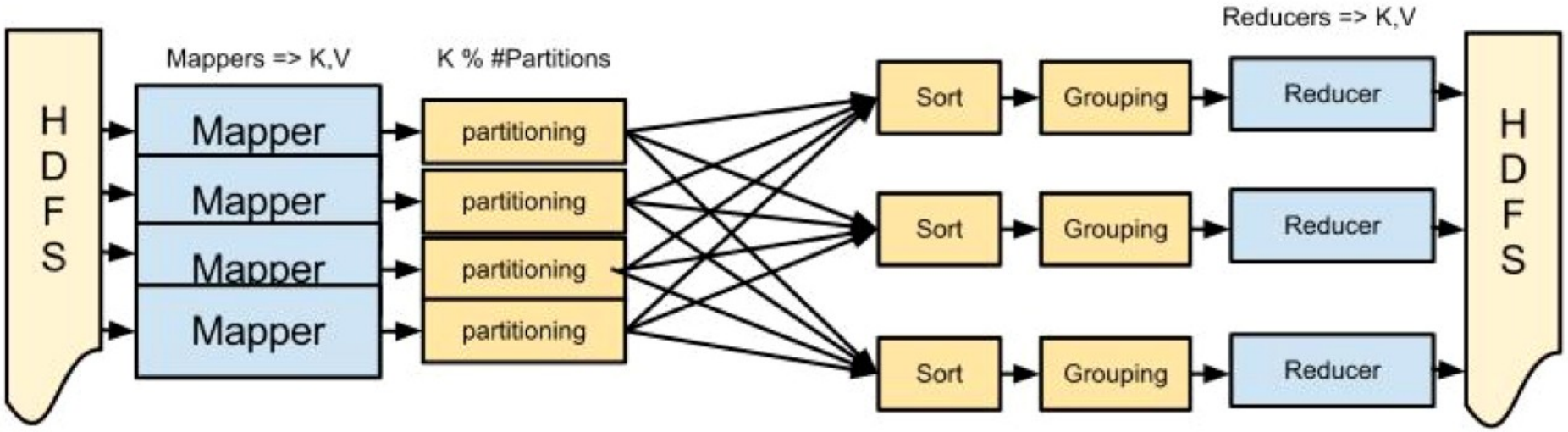
Review MapReduce

- Telah dibahas pada Sesi 19
- Didesain untuk memproses dataset besar
- Khusus masalah yang bisa terdistribusi/paralel
- Proses disebar ke banyak node secara paralel
- Tidak boleh ada dependencies/ketergantungan data dan proses

Review MapReduce

- Map
  - Master node membagi/partisi input ke sub-problem lebih kecil
  - Distribusi sub-problem ke worker node
- Reduce
  - Master node mengambil jawaban/hasil dari semua sub-problem
  - Menggabungkan semuanya (reduction) → hasil/output
- Map dan Reduce ← distributed processing

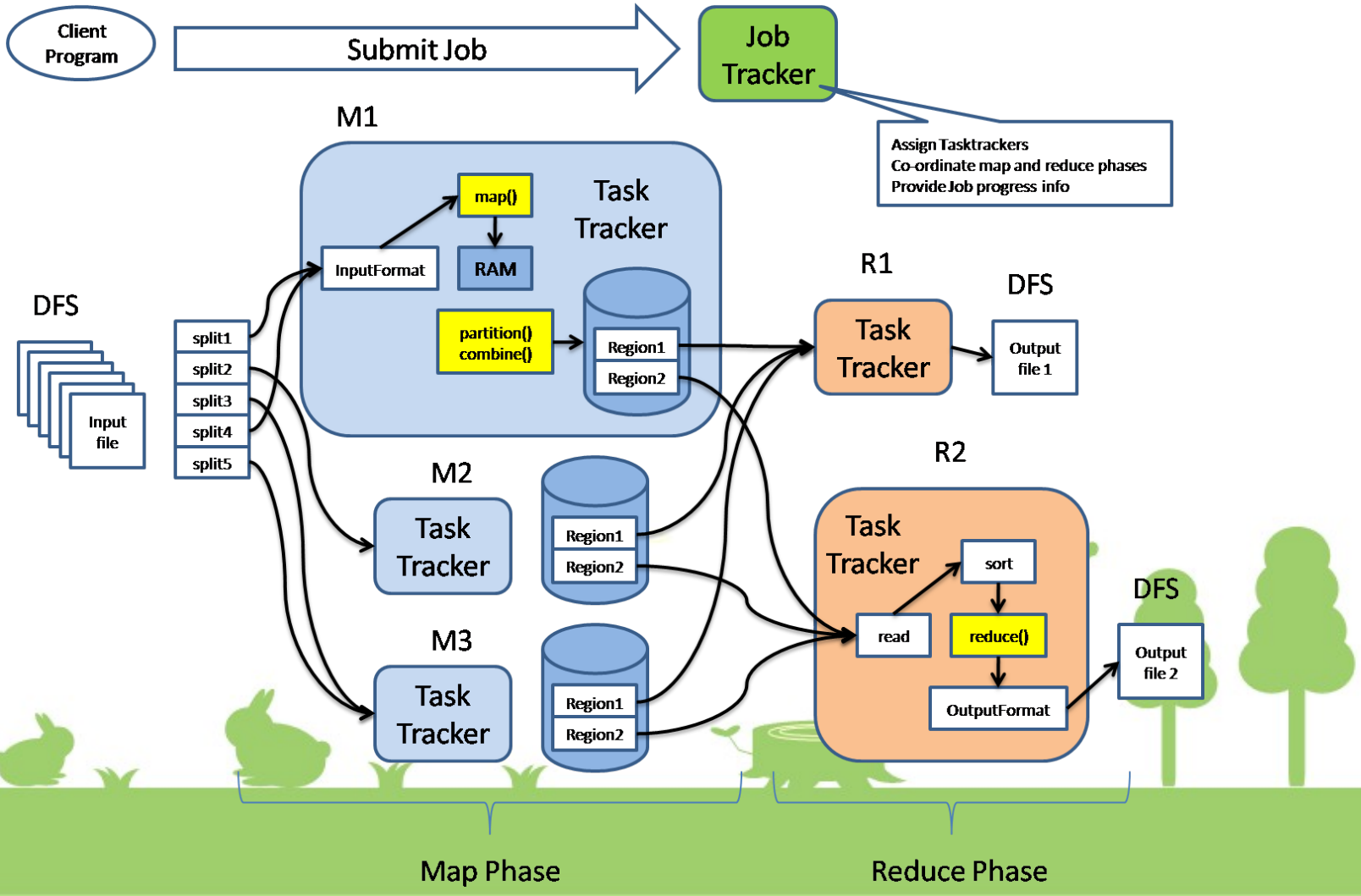
Proses MapReduce



The MapReduce Pipeline

A mapper receives (Key, Value) & outputs (Key, Value)
A reducer receives (Key, Iterable[Value]) and outputs (Key, Value)
Partitioning / Sorting / Grouping provides the Iterable[Value] & Scaling

Sebuah job dengan 1 step map dan 1 step reduce



List Proses - Map

- Map step
  - Input split Mengambil subset (sebagian) data input dari full dataset
  - Operasi dilakukan ke tiap baris dari input split (tergantung operasinya, parsing dll)
  - Outputnya di-buffered dalam memori dan taruh ke disk
  - Di-sort dan dipartisi oleh key dengan default partitioner
  - Merge sort urut tiap partisi
  - Bisa ada beberapa map secara paralel proses input split beda

List Proses - Reduce

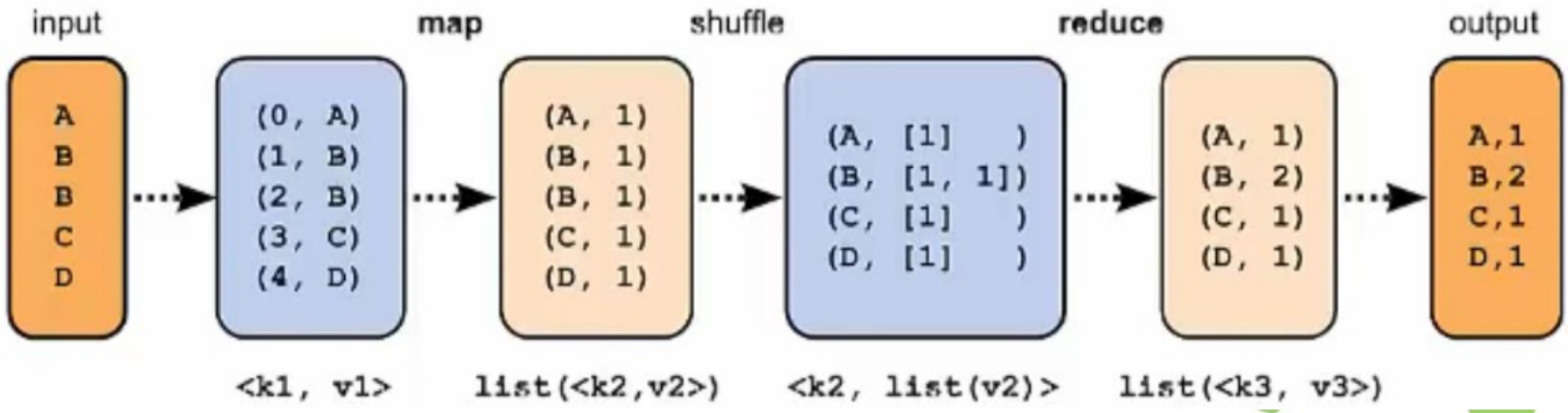
- Reduce step
  - Partisi dari output map dikocok (shuffle) ke reducers
    Partisi 1 ke reducer 1
  - Jika ada beberapa map, semua partisi 1 ke reducer 1
    Partisi 2 ke reducer 2, dst
  - Melakukan proses merge (gabung) sesuai dengan code
    contoh: jumlah kemunculan tiap karakter string
  - Hasilnya diurutkan di tiap reducer

Fundamental Data Type

- Data MapReduce yang masuk dan keluar dalam bentuk unstructured
- Sebelum masuk ke Hadoop, diubah ke key-value pair
  Hadoop menyuplai key-nya
  - Key-value
  - List

	Input	Output
Map	<k1, v1>	List(<k2, v2>)
Reduce	<k2, list(v2)>	List(<k3, v3>)

Contoh Key-Value dan List



Word Counter