Arquitectura i tecnologies del software

Pràctica (I) - BigData

**Enginyeria Informàtica**

**Escola d’Enginyeria (EE)**

**Universitat Autònoma de Barcelona (UAB)**

**Adonis González Godoy**

**Alex García Carballo**

**Contenido**

# 

# 1 Introducción

El objetivo principal que se propone es desarrollar una pequeña aplicación que implemente el algoritmo Map-Reduce desde 0, sin usar ninguna librería externa que implemente este proceso, esta aplicación tiene que contar las palabras de uno o varios ficheros que se le puedan pasar como parámetro.

También se tiene que programar un contador de palabras sin la implementación de ningún algoritmo para poder comparar los resultados de tiempo y precisión con respecto al contador de palabras con Map-Reduce.

# 2 Entorno de trabajo

El lenguaje de programación que utilizaremos para desarrollar la aplicación es **PYTHON 2.7**, hemos decidido usar este lenguaje ya que tiene una sintaxis muy legible e intuitiva que nos permite manejar listas y diccionarios de una manera más fácil, de tal manera que nos podemos centrar en el algoritmo y no tanto en la sintaxis.

# 3 WordCount

Es una pequeña y sencilla aplicación que cuenta el número de palabras dado uno o varios ficheros, adjuntamos el fichero que implementa esta funcionalidad. Descripción del fichero y cuanto tarda.

# 4 Map-Reduce

Es un algoritmo que propone resolver problemas cuando se tiene un volumen de datos grandes, con una técnica de manipulación de datos que consiste en dividir las tareas en diferentes algoritmos procesándolos de forma paralela, está basado en dos fases el **map** y el **reduce**.

A continuación, mostramos un diagrama donde podemos observar la implementación grafica del Wordcount implementando el Map-Reduce.

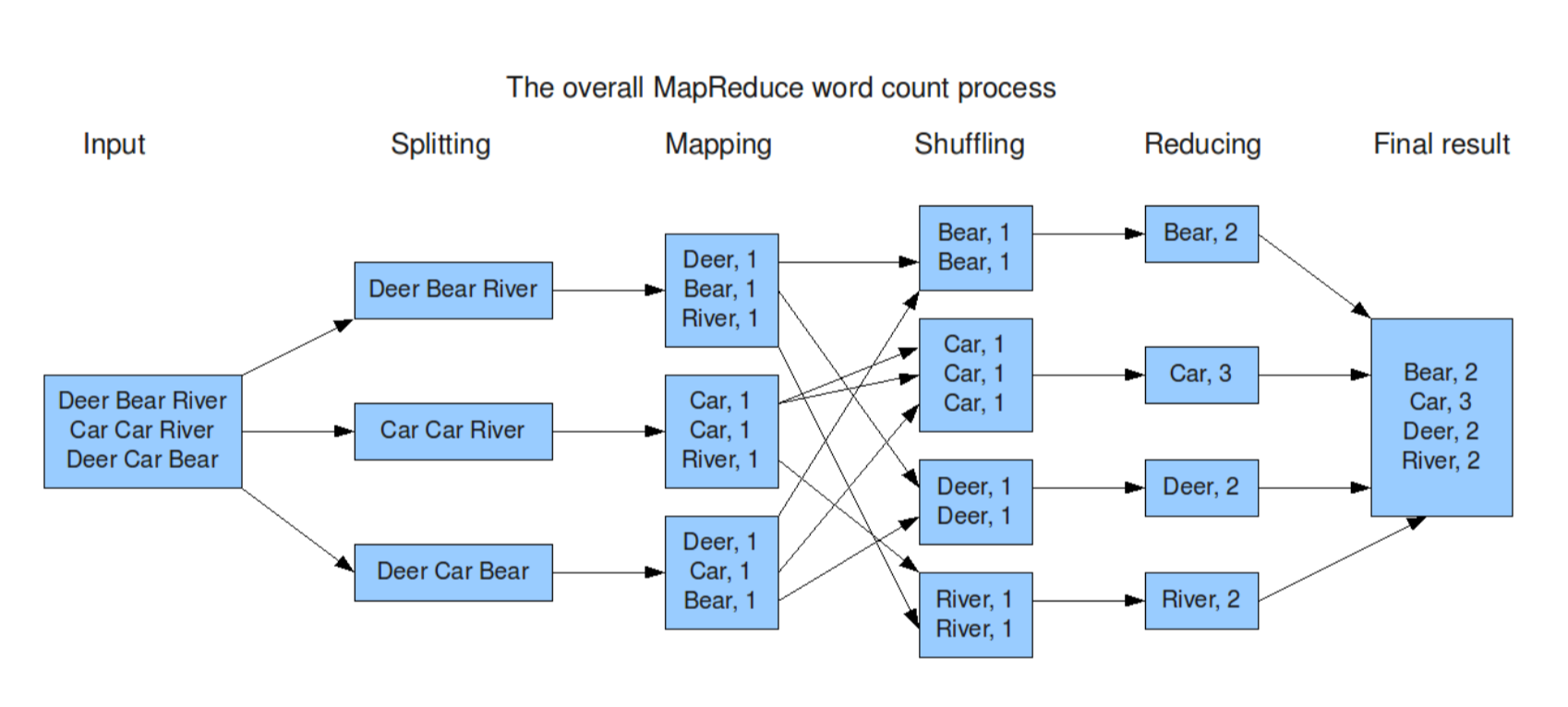
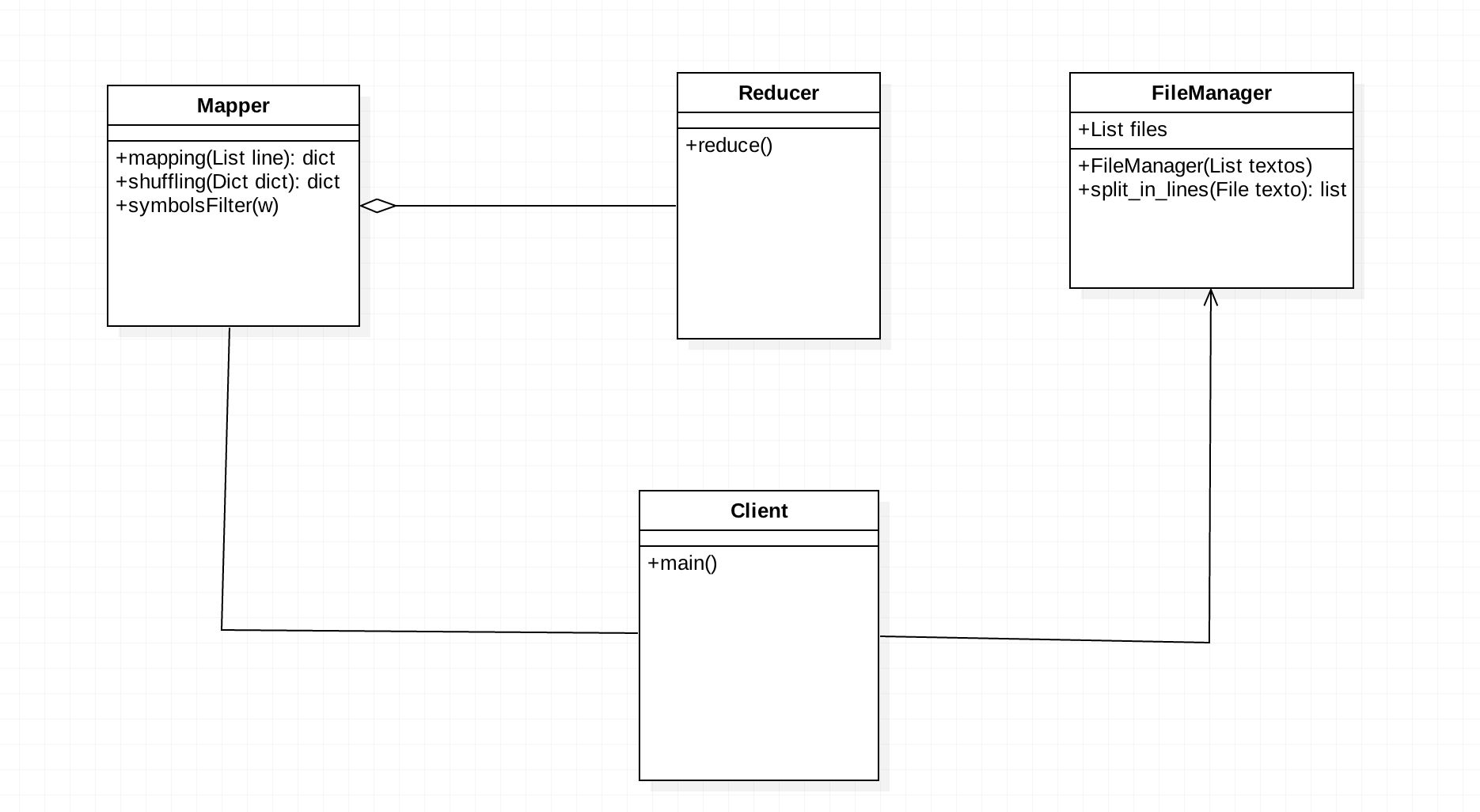
****

Diagrama - Map-Reduce

# 5 Implementación

Para definir la estructura de la aplicación tenemos el siguiente diagrama UML.

## UML



EL UML falta añadir cosas que no lo tenia tan claro.

# 6 Diagrama de secuencias

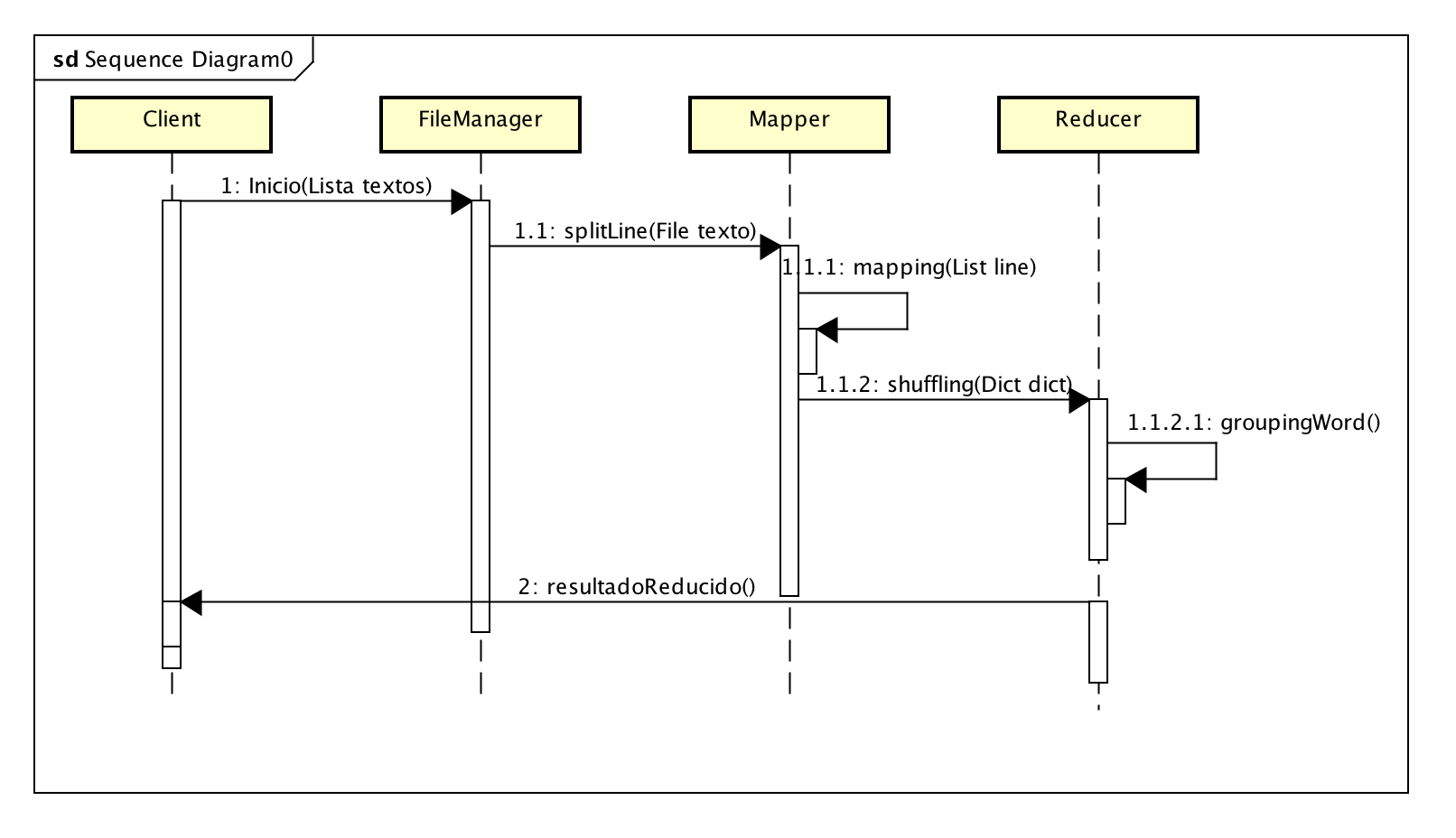


Diagrama - Secuencias

# Funcionamiento del programa

La clase **FileManager** gestiona los ficheros que se pasarán por parámetro, también se encarga de dividir los ficheros en líneas, esta clase equivale a la fase de separar o splitting. Todas estas líneas de uno o varios ficheros se guardarán en una lista.

La clase **Mapper** contiene tres funciones:

symbolsFilter(w) 🡪 se encarga de quitar los símbolos a las palabras

mapping() 🡪 creamos un diccionario donde tendremos dos valores, una palabra y una valor clave para cada una de estas, en concreto el valor 1. Esta función acaba retornando un diccionario de palabras con su valor.

shuffle()

# Resultados

Graficas con resultados de distinas ejecuciones.

# Compilación

Para ejecutar el programa debemos tener instalado la versión de Python 2.7.

Usaremos el comando siguiente:

Python mapReduce.py –file1 filen

Se puede pasar uno o varios ficheros por parámetros.