

5. Para emular MapReduce utilizando activeMQ, yo pensaría en cómo llevar adelante todas las etapas de MapReduce contando únicamente con el código de las funciones `map()` y `reduce()` provistas por el usuario.

1. **Lectura de archivos:** para hacer la lectura de las (key, value) en paralelo, desde un proceso principal haría spawn de procesos que escuchen en una cola. Luego enviaría mensajes a la cola indicando que parte de los datos ese proceso tiene que leer de disco.

Estos mensajes se entregarían desde un broker de manera round-robin hacia los procesos. De esta forma utilizo el direccionamiento remoto que me dan las colas para coordinar los procesos que podrían estar distribuidos en un cluster.

2. **Map:** en cada proceso de los anteriores corro una o varias veces el `map()` que fue provisto por el usuario sobre los datos que leyo cada proceso. Digo una o varias veces porque probablemente cada proceso lea los datos de más de una llave.

3. **Combine:** en la etapa de combine convierto los (key, value) en (key, value[]) agrupando las llaves iguales. Aquí puedo usar un topic por cada key y crear nuevos procesos que estén suscriptos a ese topic.

Los procesos maps del punto anterior emiten pares (key, value), desde estos procesos emito a un broker utilizando cada key distinta como tópico. Los reducers estarían suscriptos a esos tópicos. Probablemente no sepa los valores de las llaves de antemano, por lo tanto se podría hacer un mapeo de llaves arbitrarias emitidas por los mappers a un conjunto predefinido.

Ejemplo, supongamos que se emiten las siguientes llaves {45, 67, 89, 12, 24}, puedo hacer un mapeo,

45 → 1

67 → 2

89 → 3

...

24 → 5

De esta manera puedo tener de antemano procesos suscriptos a los tópicos 1, 2, 3, ..5 y continuar el procesamiento de `reduce()`.

Podría tener alguna especie de redundancia instanciando múltiples procesos que escuchen el mismo tópico y tomar el resultado del que termine primero.

4. **Reduce:** dentro de los mismos procesos del punto anterior, de alguna manera marco la finalización del combine anterior (por ahí con algún mensaje). Con esto me refiero a que en algún momento el que se suscribe al resultado del combine tiene que saber que para la llave que está escuchando ya no van a llegar más valores.

Luego ejecuto las funciones del usuario sobre los datos que agrupé. En estos mismos procesos escribo a disco.