

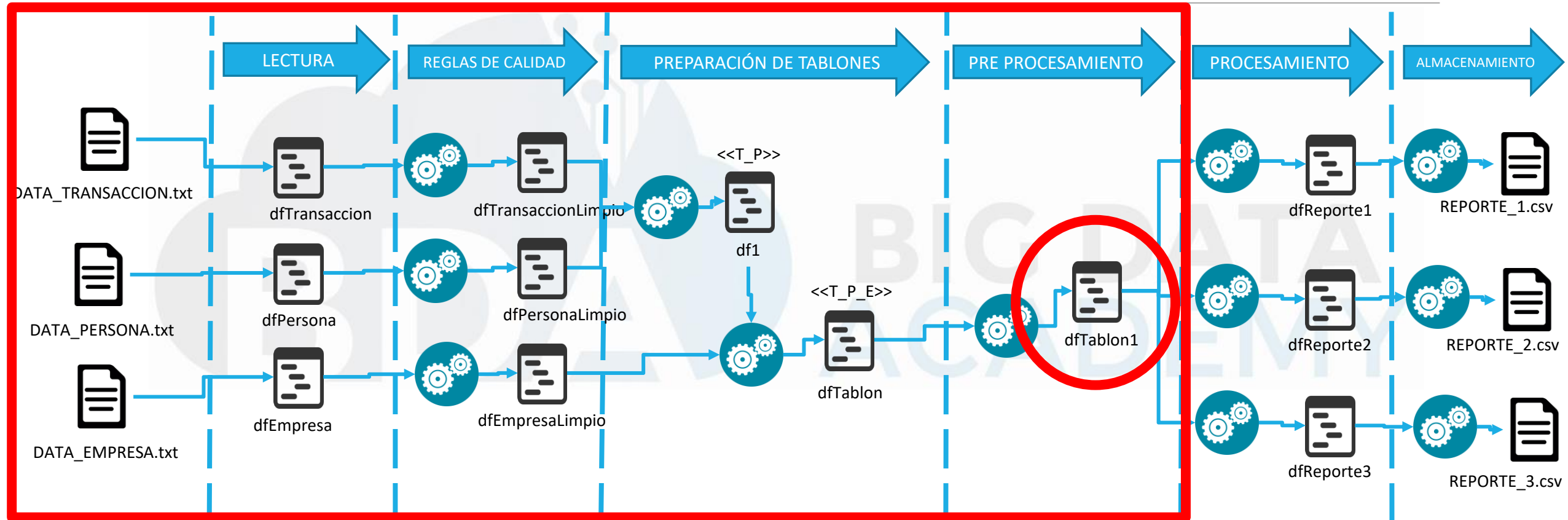
---

# Patrón de Diseño CACHE

---

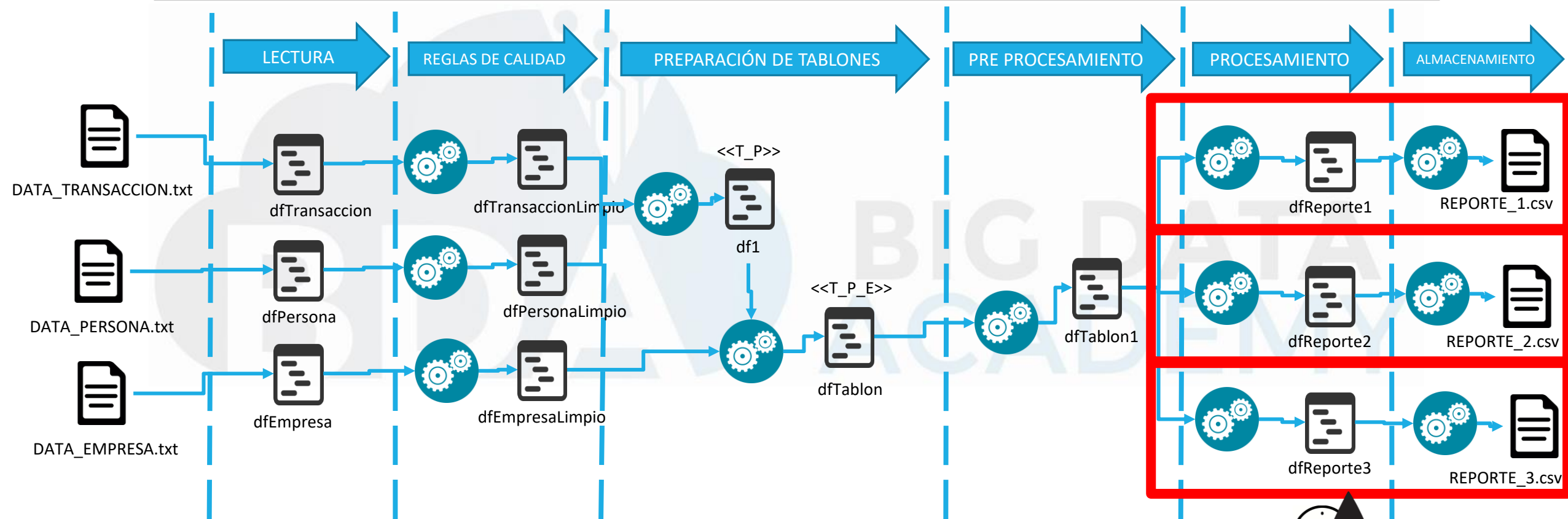
# Proceso a implementar

Supongamos que esta parte del proceso, hasta la creación del “dfTablon1”, **toma 5 horas en ejecutarse**

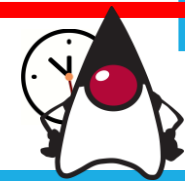


# Proceso a implementar

Supongamos que la escritura de los tres reportes, **toma 10 minutos en ejecutarse cada uno**

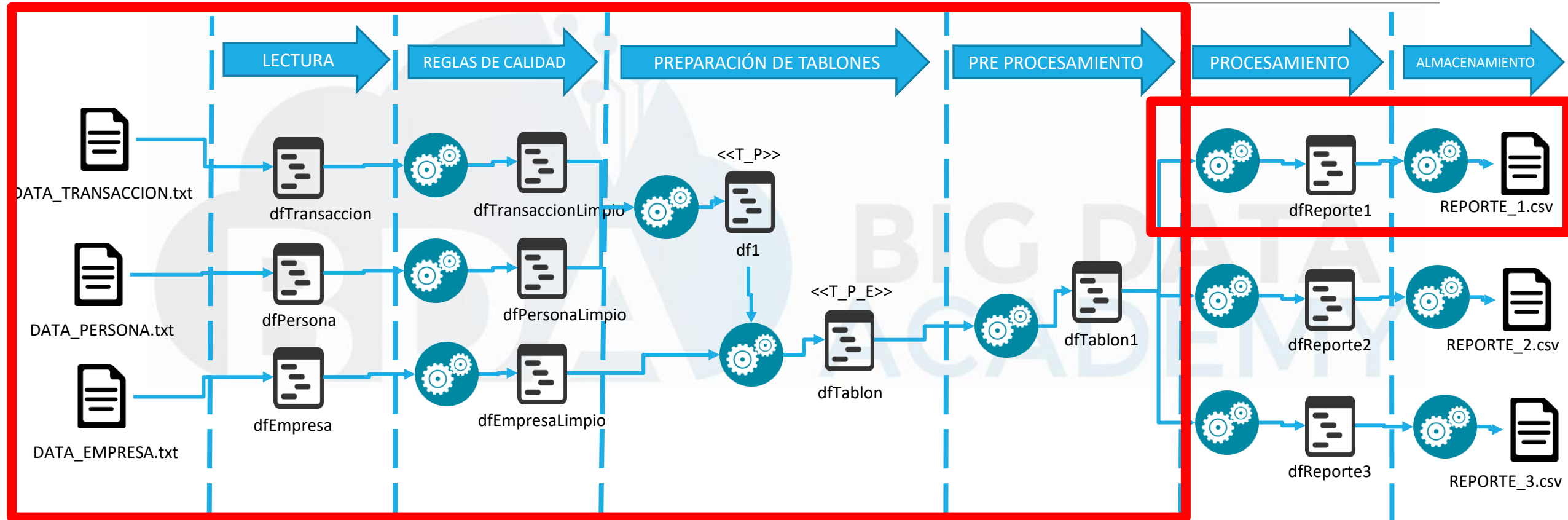


**¿El proceso en total demoró 5 horas y 30 minutos?, no necesariamente, depende del GC**



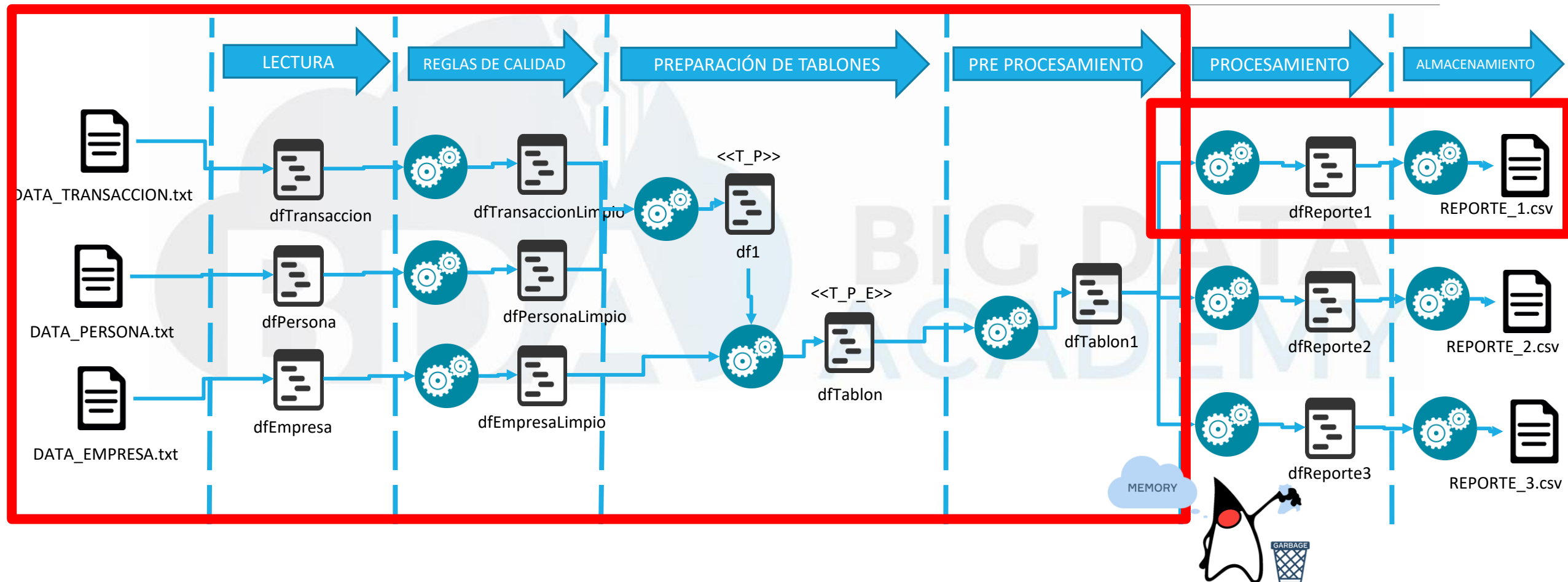
# Primer “action” que se ejecuta

Primero se ejecuta el action “write” del “dfReporte1”, **esta parte del proceso toma 5 horas y 10 minutos**



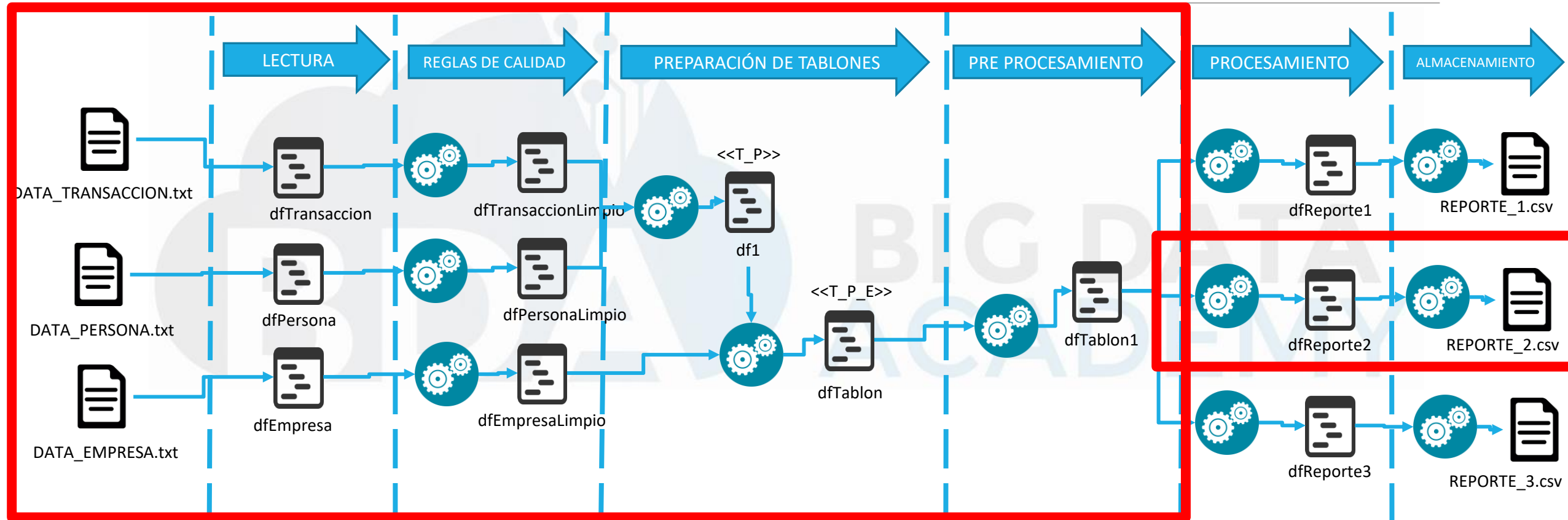
# El peor de los casos

El Garbage Collector se encuentra en esa zona de memoria RAM, y luego de que el “action” finaliza, **borra todos los dataframes de memoria RAM**



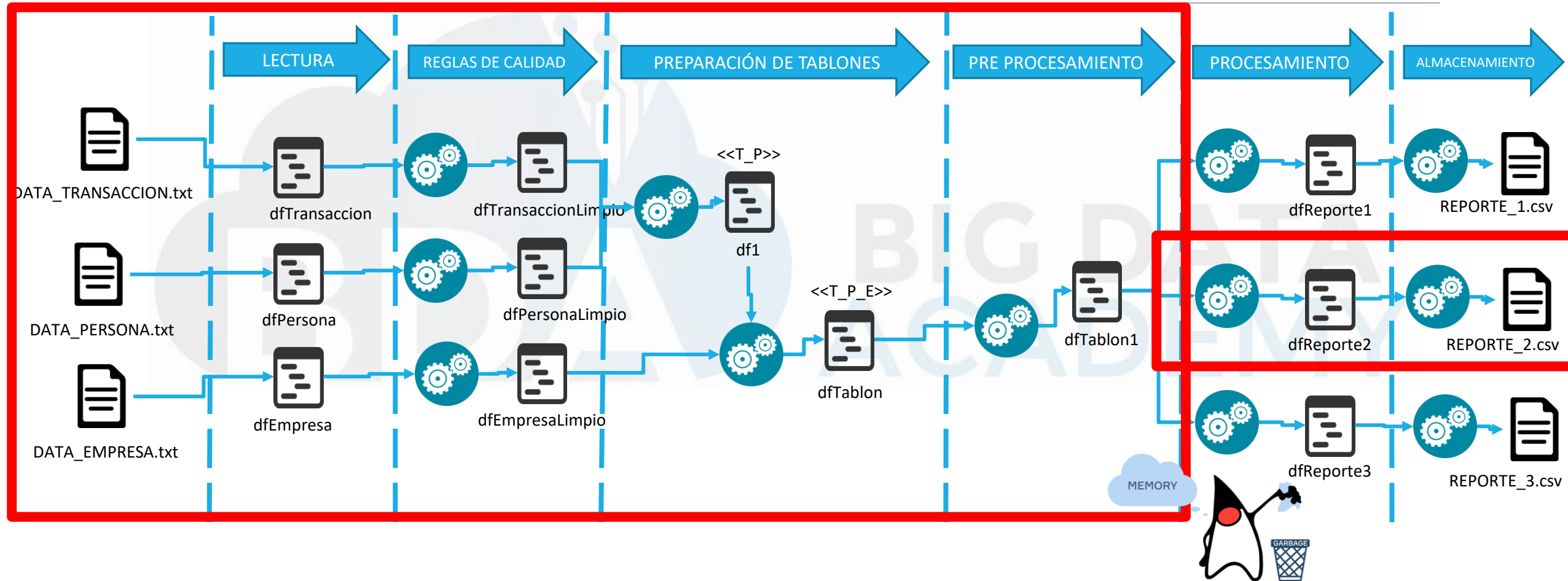
# Segundo “action” que se ejecuta

Se ejecuta el action “write” del “dfReporte2”, **como no hay nada en la RAM, tenemos que esperar nuevamente 5 horas para que se cree el “dfTablon1” y 10 minutos para el “dfReporte2”**



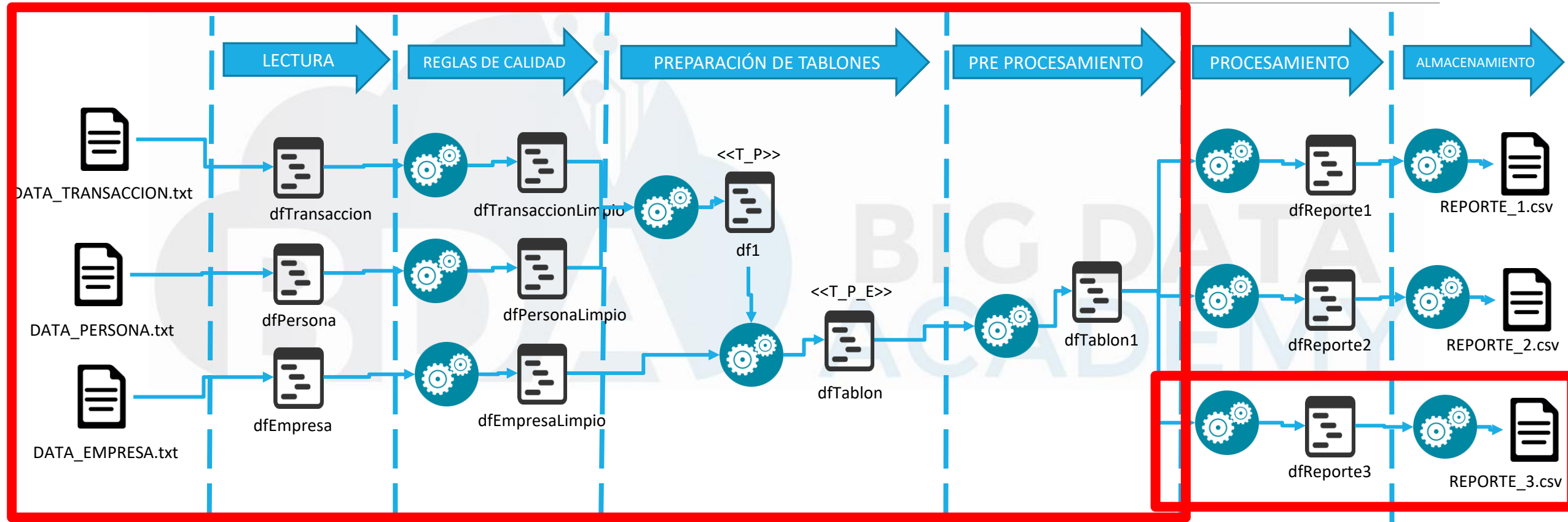
# Nuevamente, el peor de los casos

El Garbage Collector se encuentra en esa zona de memoria RAM, y luego de que el “action” finaliza, **borra todos los dataframes de memoria RAM**



# Tercer “action” que se ejecuta

Se ejecuta el action “write” del “dfReporte3”, **como no hay nada en la RAM, tenemos que esperar nuevamente 5 horas para que se cree el “dfTablon1” y 10 minutos para el “dfReporte3”**





# Tiempo total de ejecución

dfReporte1: 5 horas + 10 minutos

dfReporte2: 5 horas + 10 minutos

dfReporte3: 5 horas + 10 minutos

**TOTAL: 15 horas + 30 minutos**

Debido al problema del Garbage Collector



¿Cómo lo solucionamos?

---

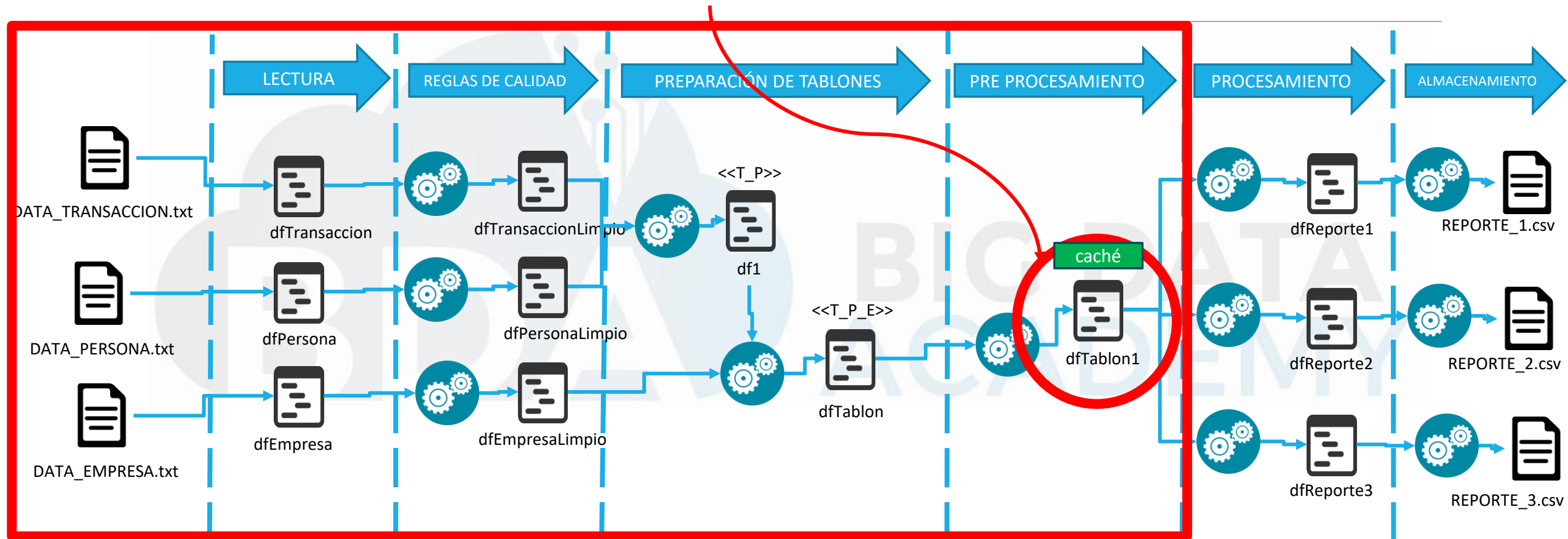
# Solución: Patrón de diseño CACHÉ

---



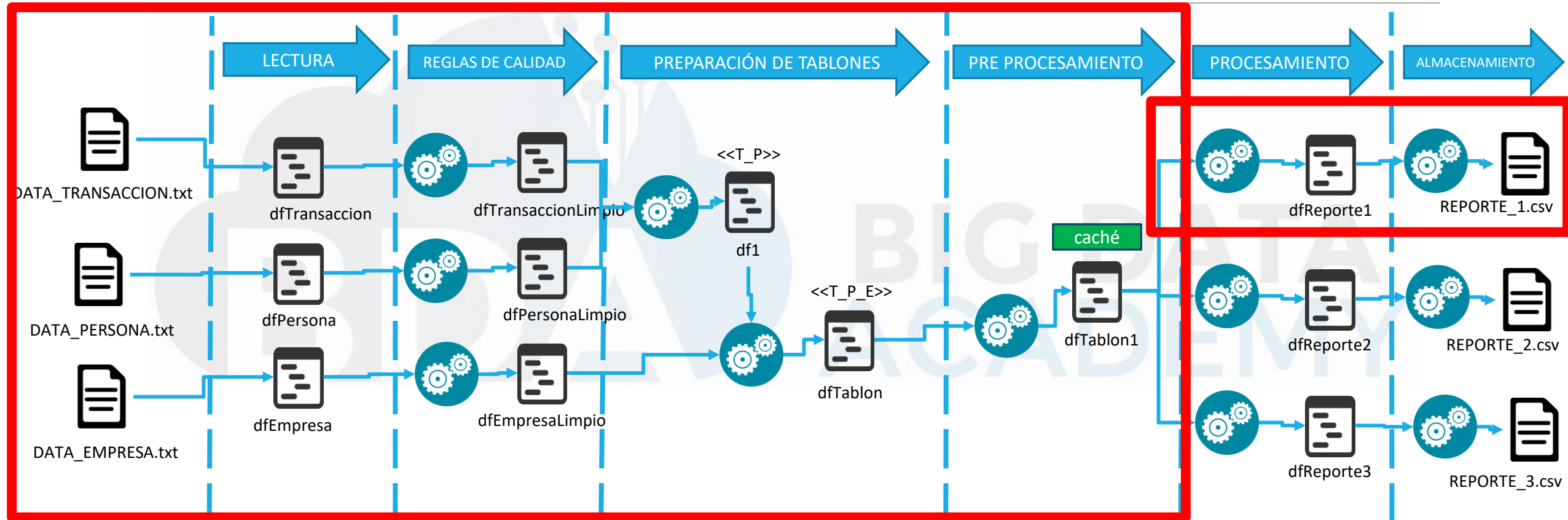
# Patrón de diseño CACHÉ

Deberemos de **marcar en la "caché" al dataframe que sea reutilizado** más de una vez para crear otros dataframes



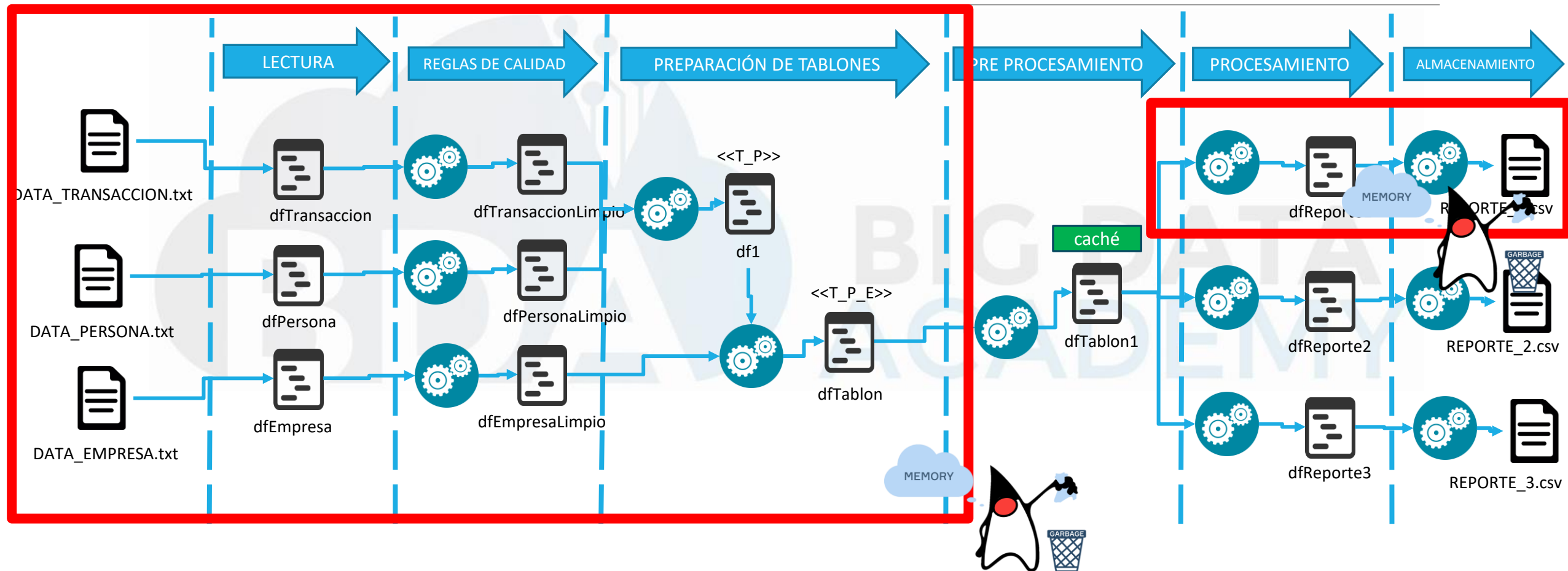
# Primer “action” que se ejecuta

Primero se ejecuta el action “write” del “dfReporte1”, **esta parte del proceso toma 5 horas y 10 minutos**



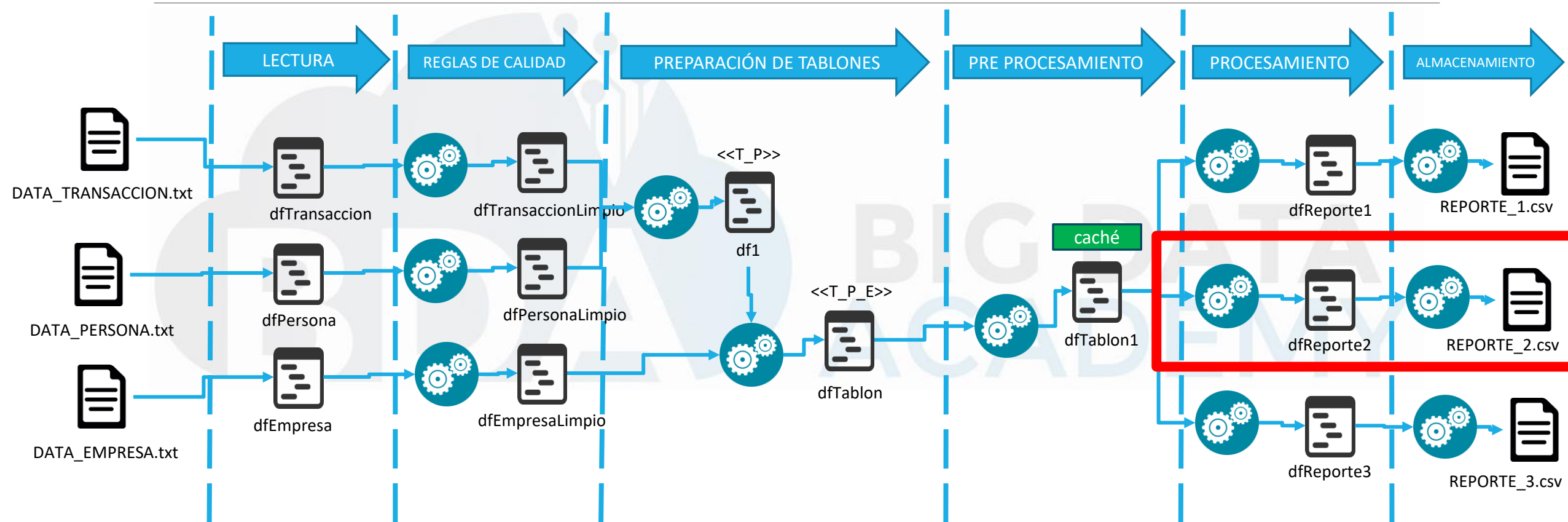
# El peor de los casos

El Garbage Collector se encuentra en esa zona de memoria RAM, **borra todos los dataframes, menos el "dfTablon1"**



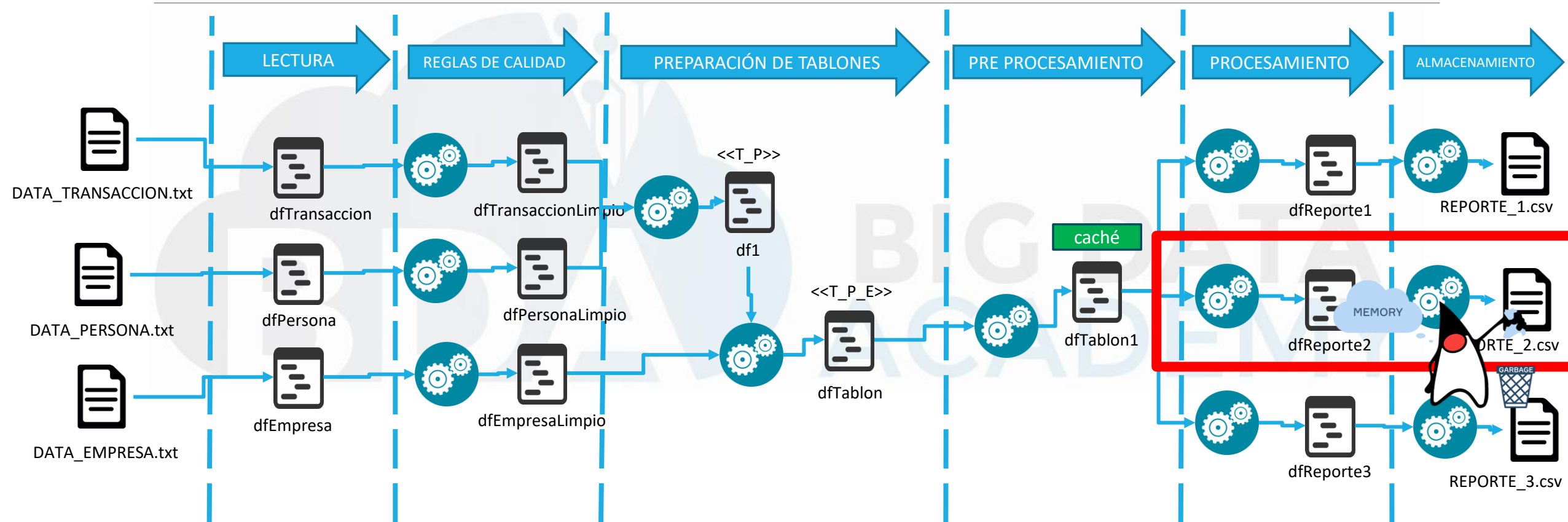
# Segundo “action” que se ejecuta

Se ejecuta el action “write” del “dfReporte2”, **como aún existe el “dfTablon1”, sólo deberemos esperar 10 minutos para que se calcule el “dfReporte2”**



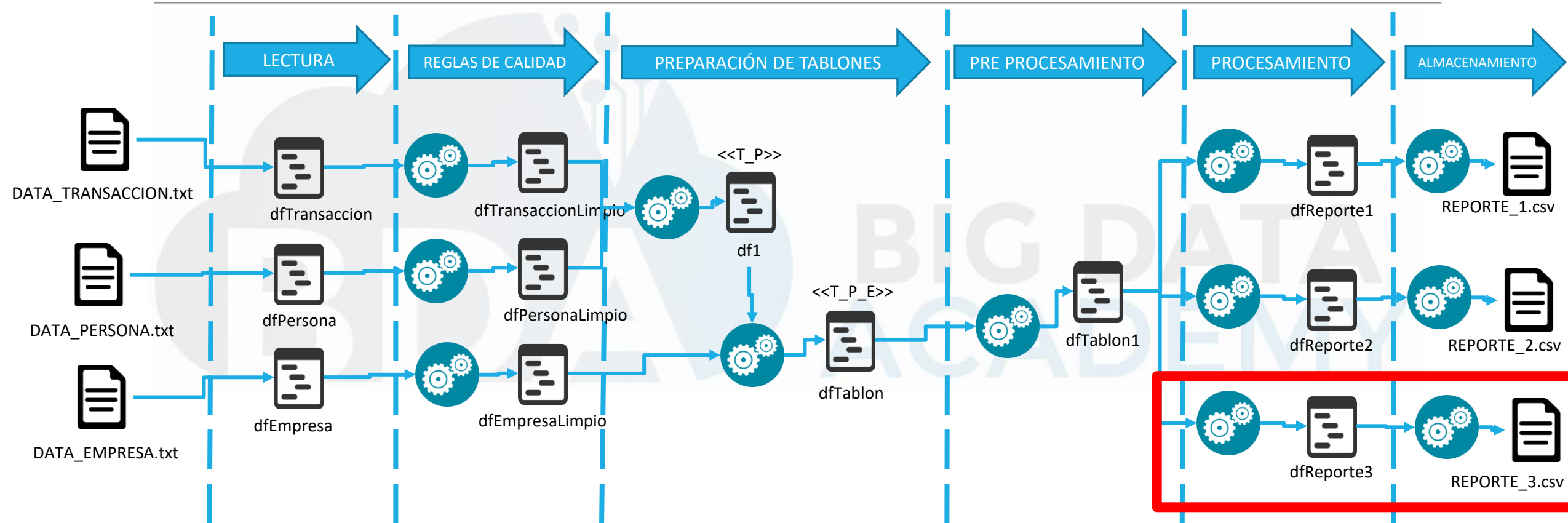
# Nuevamente, el peor de los casos

El Garbage Collector se encuentra en esa zona de memoria RAM, **borra todos los dataframes, menos el "dfTablon1"**



# Tercer “action” que se ejecuta

Se ejecuta el action “write” del “dfReporte3”, **como aún existe el “dfTablon1”, sólo deberemos esperar 10 minutos para que se calcule el “dfReporte3”**





# Tiempo total de ejecución

dfReporte1: 5 horas + 10 minutos

dfReporte2: 10 minutos

dfReporte3: 10 minutos

**TOTAL: 5 horas + 30 minutos**

El proceso se ejecutó 10 horas más rápido

¿Cuándo usar el patrón de diseño “CACHE”?: **Si un dataframe será usado para calcular dos o más dataframes**, debemos marcarlo en la caché