

Patrón de diseño Maestro – Esclavo

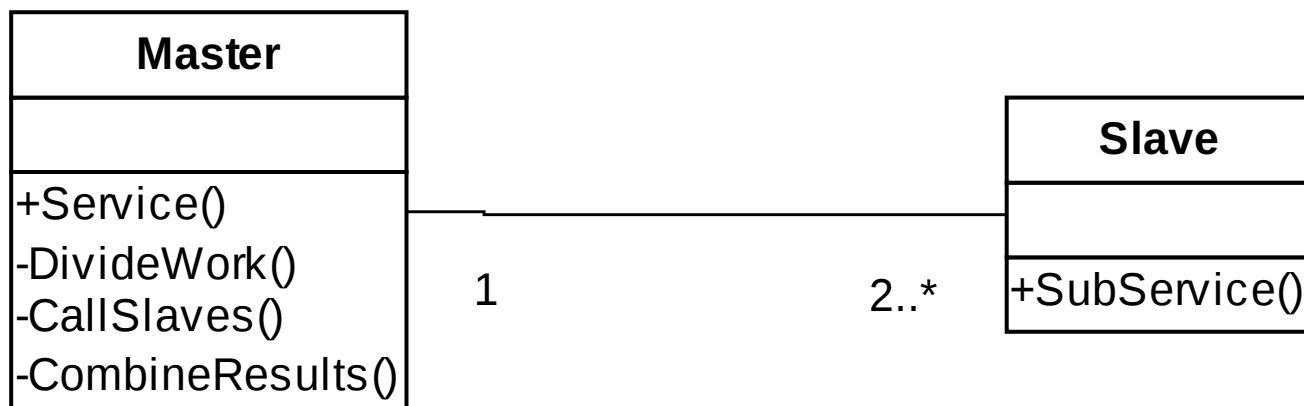
José Antonio Martínez Torres
<http://www.antoniomtz.org>

Problema

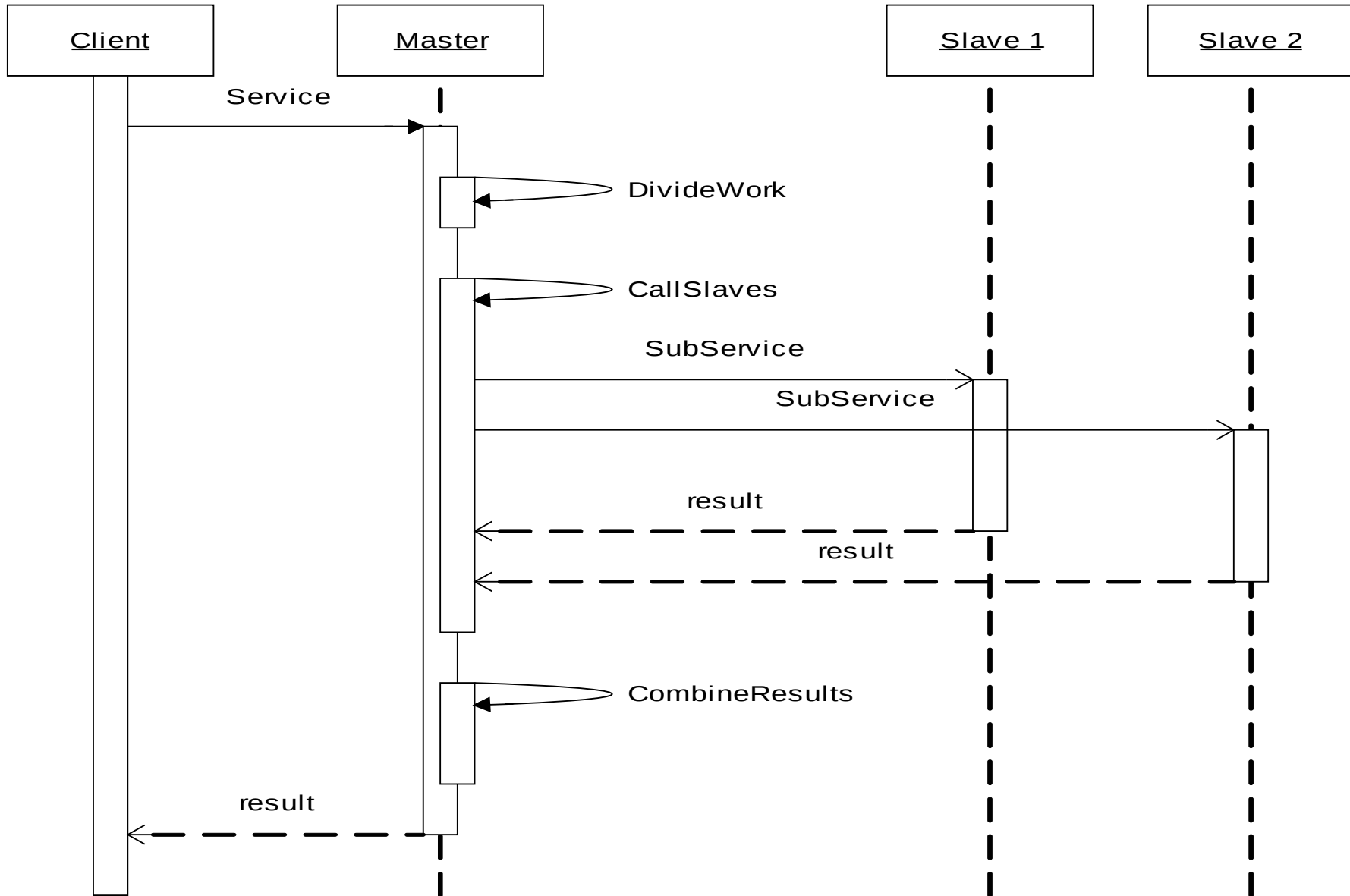
- Un sistema debe ser **Fault-Tolerant** (ej. Que el sistema siga funcionando, incluso cuando algunos de sus componentes fallen)
 - Ej. un sitio e-commerce debe seguir funcionando incluso cuando un servidor de base de datos se cae.
- Un sistema debe producir **resultados precisos**, incluso cuando algunos de sus componentes fallen.
 - Ej. Un software de control de avión, debe de producir resultados precisos, incluso cuando alguna pieza de hardware falle.

Solución

- Un componente maestro, divide su trabajo en subtareas - **Divide y vencerás!**-
- El maestro asigna subtareas a múltiples esclavos.
- Los esclavos calculan sus subtareas en **paralelo** y **retornan** sus resultados parciales al maestro.
- El maestro calcula el resultado final basado en los resultados parciales que fueron regresados por los esclavos.



Comportamiento dinámico



Implementación

- La implementación puede ser relativamente compleja.
- Los esclavos se implementan como hilos ó procesos separados.
- Maestro-esclavo pueden correr sobre una sola máquina ó también pueden ser distribuidos en varias máquinas (Ej. Cluster)

Consecuencias

- Maestro-esclavo no siempre es factible porque algunas tareas no pueden ser particionadas.
- Difícil de implementar
- Dificultad en la portabilidad, si se tiene una fuerte dependencia sobre el hardware al que fué configurado.