

Memoria De la AA1

David Auladell
Maurizio Carlotta
Abraham Armas Cordero

2º CDI(dev) Tarda

Jerarquía de procesos	2
Tabla de file Descriptors y Explicación	3
Comunicación entre procesos	3
Conclusiones	3

Jerarquía de procesos

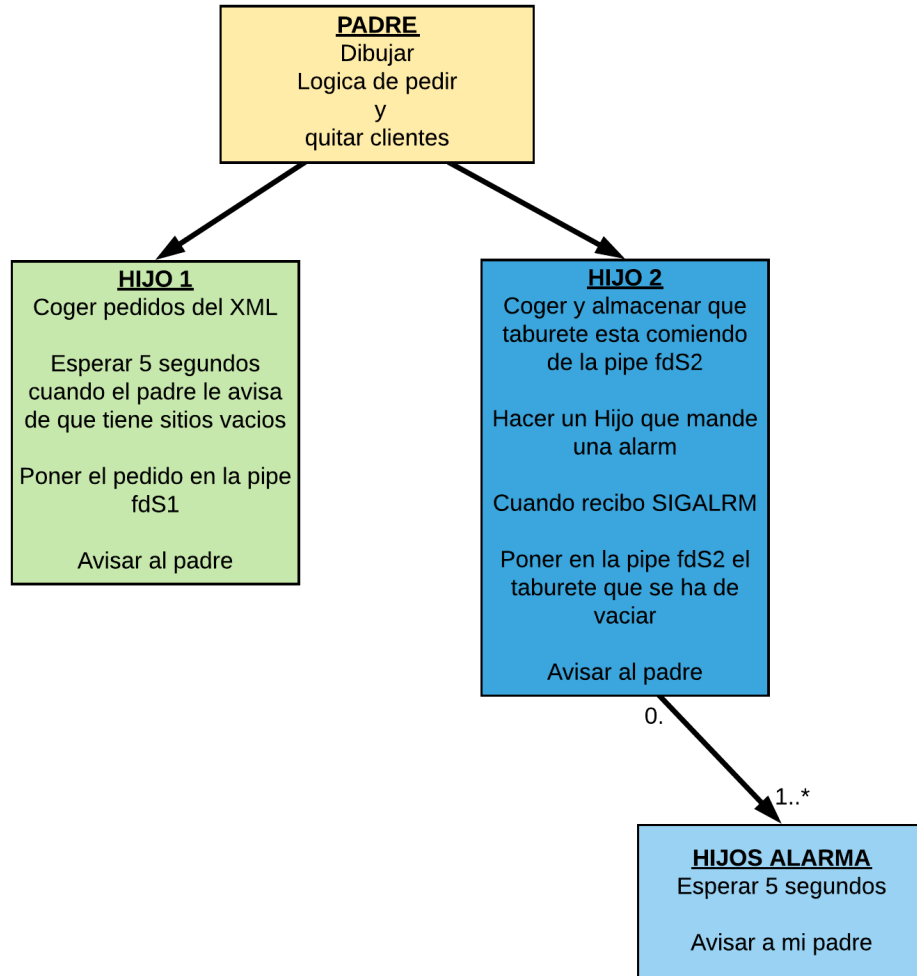
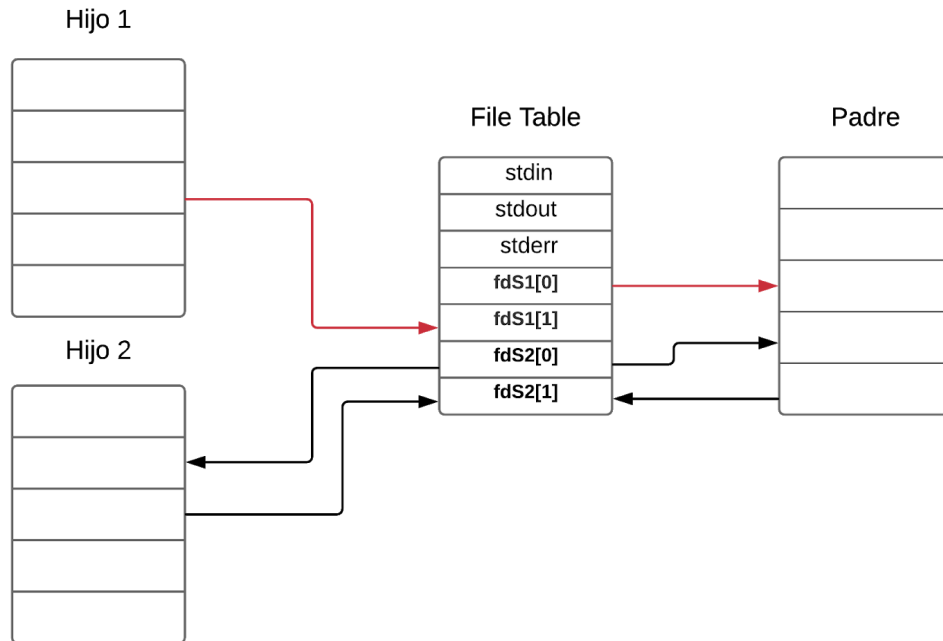


Tabla de file Descriptors y Explicación

La línea roja es para diferenciar la pipe del Hijo 1 sobre la del Hijo 2



- **fdS1:** Unidireccional
 - `fdS1[1]`: Sirve para que el hijo uno le diga al padre de que color es el pedido
 - `fdS1[0]`: Lee el color en formato RGB como string
- **fdS2:** Bidireccional
 - `fdS2[1]`:
 - Padre: Le envía al hijo que taburete empieza a comer
 - Hijo: Le envia al Padre que taburete a terminado de comer
 - `fdS2[0]`:
 - Hijo: Recoge que taburete está comiendo.
 - Padre: Recoge que taburete ha terminado de comer.

Comunicación entre procesos

NOTA: todas las líneas de código están en el main.cpp y con este formato **(10)**

Para conseguir clientes:

1. El padre envía un signal SIGUSR1 al Hijo 1 cuando tiene taburetes vacíos**(216)**
2. El hijo con este SIGUSR1 empieza TriggerAlarm() que le hace esperar 5s y después envía un SIGALRM**(71)**
3. Con este SIGALRM hace la función CargarClienteSon()**(72)** que escribe en el fdS1 una string con el color del pedido**(290)**, lo quita del contenedor de pedidos. después manda un signal SIGUSR1 al padre**(291)**.
4. El padre con el SIGUSR1 ejecuta CargarClienteFath()**(129)** que lee la pipe fdS1 parsea la string a un sf::Color **(409)** y pinta al nuevo cliente con DrawClient() **(328)**

Para quitar clientes:

1. El padre escribe en la pipe fdS2 el número del cliente que ha empezado a comer**(195)** y manda un SIGUSR2 al Hijo 2**(196)**.
2. El Hijo 2 con SIGUSR2 ejecuta la función UnLoadWhatClient()**(119)** que lee la pipe fdS2 y guarda el número en una cola**(305)**. Después crea un proceso hijo que espera 5 segundos y le devuelve un SIGALRM**(310)**.
3. Con el SIGALRM ejecuta la función UnLoadClient()**(118)** que escribe en la pipe fdS2 el número del cliente que hay que quitar**(321)** y lo quita de la cola. A continuación manda un SIGUSR2 al Padre**(324)**
4. El padre con SIGUSR2 ejecuta la función VaciarMesa()**(130)** que lee la pipe fdS2 que taburete hay que vaciar**(388)**, vacía el taburete y el pedido después decrementa el número de clientes.

Conclusiones

Bueno en cuanto a dificultad ha sido bastante sencilla de entender la práctica estaba bien definida y creemos que ha conseguido su objetivo de asentar lo aprendido en clase.

Hemos tenido fallos de todo tipo pero el fallo que resaltaría porque tiene que ver con el temario sería que se nos olvidó que si hacíamos un alarm en un proceso y no das tiempo a que se termine el alarm y la vuelves a llamar se resetea y hacía que los clientes se fueran en tiempos que no tocaban o no se fuesen, lo solucionamos haciendo que el hijo creará otros hijos que se encargaran de esperar y justo cuando terminan enviar un signal directo que ejecutase la función adecuada