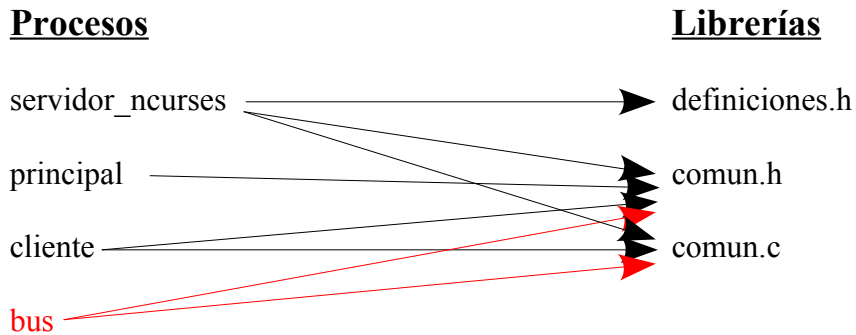


Índice

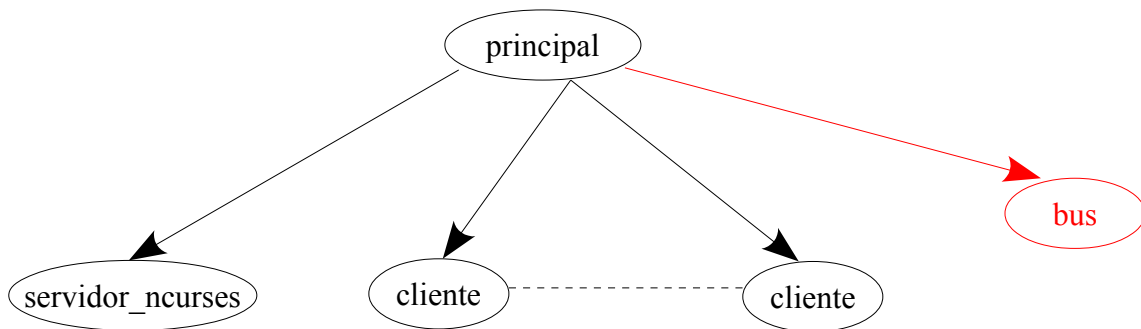
1. Ficheros.....	1
2. Esquema de creación.....	1
3. Esquema de sincronización.....	1
4. Esquema de comunicación.....	2
5. Funciones especiales.....	2
6. Procesos.....	3

1. Ficheros



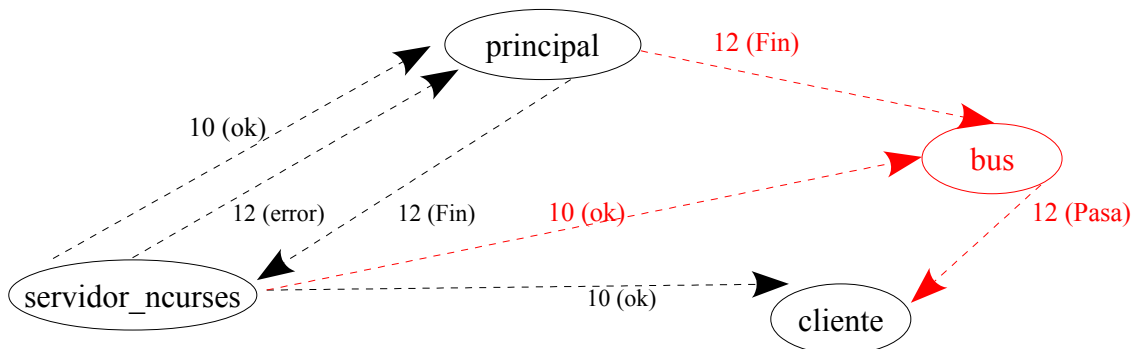
fichcola.txt → Para crear cola de mensajes

2. Esquema de creación

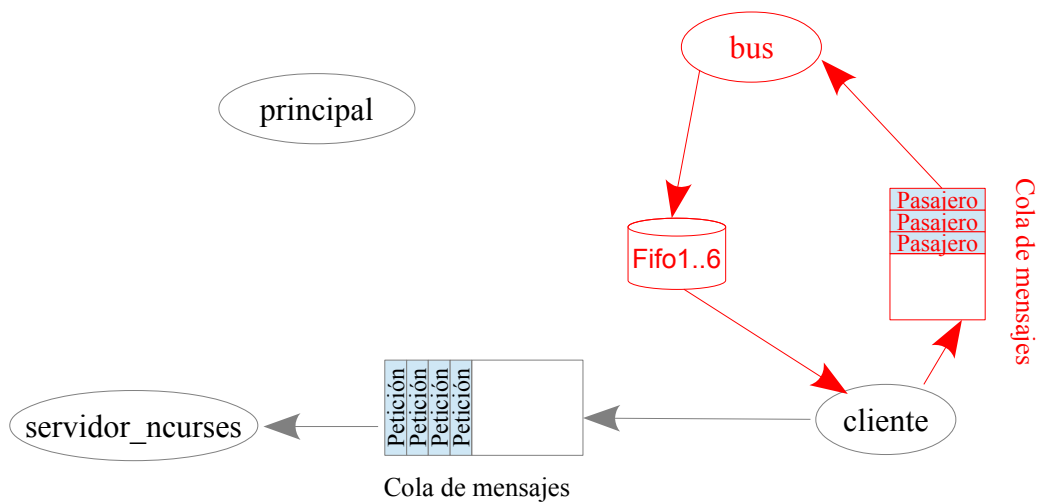


Bucle desde 1 hasta 30 (maxclientes)

3. Esquema de sincronización



4. Esquema de comunicación



5. Funciones especiales

Función **pinta** (Va en bus.c)

```
void pinta(int cola, int parada)
{
    struct tipo_elemento peticion;

    peticion.tipo=1; //Los clientes son tipo 2, el autobus tipo 1
    peticion.pid=getpid();
    peticion.parada=parada; //Numero de parada.12 Entre la 1 y la 2
    peticion.inout=0; //no se usa
    peticion.pintaborra=0; // no se usa
    peticion.destino=0; //no se usa

    msgsnd (cola, (struct tipo_elemento *) &peticion,sizeof(struct
    tipo_elemento)-sizeof(long),0);

    if(!llegal0) pause();//Espero conformidad de que me han pintado,
        llegal0=0;          //sino me mataran
}
```

6. Procesos

principal.c

```
main()
{
    Prepararse para recibir señales 10 y 12;
    Crear servidor_ncurses con creaservigraf;
    Inicializamos semilla;
    Esperar señal de ok o de error del servidor gráfico;
    Creamos las fifos;
    Creamos el bus;
    for (i=1;i<=MAX_CLIENTES;i++)
    {
        Crear cliente con creaproceso;
        Espera aleatoria antes de crear otro cliente;
    }
    Espera fin de sus hijos;
    Avisa al bus de fin con señal 12;
    Avisa a servidor grafico con señal 12;
    Cerramos y borramos las fifos;
}
```

cliente.c

```
main()
{
    Prepararse para recibir señal 10;
    Prepararse para recibir señal 12;
    Crear la cola de mensajes;
    Crear cola de mensajes entre bus y cliente;
    Inicializamos semilla;
    Generar parada de llegada y de bajada distintas aleatorias;
    Abre fifo de la parada de bajada;
    Se visualiza en la parada de llegada;
    Escribe en la cola de mensajes entre bus y cliente;
    Espera llegada del bus (señal 12);
    Se borra;
    Se pinta en el bus;
    Espera testigo de parada de bajada;
    Se borra del bus;
    Se visualiza en la parada de bajada;
}
```

bus.c

```
main()
{
    Prepararse para recibir señal 10 y 12;
    Inicializamos semilla;
    Indicamos numero de asientos libres;
    Crear las dos colas de mensajes;
    Abrimos todas las fifos;
    Inicializamos el numero de montados de cada parada;
    while (1)
    {
        for (parada=1;parada<=NUMPARADAS;parada++)
        {
            Se pinta el bus en la parada;
            Bajamos a la gente de esa parada;
            Indicamos que no hay nadie montado en esa parada;
            Montamos a la gente de esa parada;
            Se pinta bus entre paradas;
            Se espera un tiempo de recorrido;
        }
    }
}
```

Abrimos todas las fifos;

```
for (i=1;i<=NUMPARADAS;i++)
{
    sprintf(nombrefifo,"fifo%d",i);
    fifos[i]=open(nombrefifo,O_WRONLY);
}
```

Inicializamos el numero de montados de cada parada;

```
for (i=1;i<=NUMPARADAS;i++) montados[i]=0;
```

Bajamos a la gente de esa parada;

```
for (gente=1;gente<=montados[parada];gente++)
{
    write(fifos[parada], &testigo, sizeof[testigo]);
    libres++;
    retardo;
}
```

Montamos a la gente de esa parada;

```
while (haya espacio en el bus && haya pasajeros esperando en
      esa parada)
{
    Avisamos al pasajero para que suba al bus;
    libres--;
    Aumentamos el numero de montados para la parada de
    bajada del pasajero que acaba de subir;
}
```

Se pinta bus entre paradas;

```
if (parada==NUMPARADAS) pinta (colagrafica,parada*10+1);
else pinta (colagrafica, parada*10+parada+1);
```