COMUNICAZIONE TRA PROCESSI > 3.2.

I processi comunicano tra di loro senza entità terze.
La comunicazione viene stabilita da un protocollo, ovvero un insieme di regole.

Tipi di Processi Concorrenti

Possono essere

- Indipendenti: Non influenzano e non sono influenzati da altri processi ⟶ Non condivide dati
- Cooperanti: Influenzano e sono influenzati. Ogni processo che condivide dati è cooperante

La cooperazione può essere necessaria per velocità e modularità.

La comunicazione avviene tramite la IPC o Inter Process Communication.

Modi di IPC

La IPC può essere implementata come:

- **Message Modelling**
  I processi in maniera attiva si scambiano messaggi.
  - Vantaggi: facile da implementare ⟶ Canale di com.
  - Svantaggi: non adatto a messaggi grandi, Impiega sempre il Kernel.
- **Shared Memory**
  Si usa un'area condivisa in cui scambiare le informazioni
  ⟶ Non passa dal Kernel
  - Vantaggi: Più veloce, adatta a messaggi grandi
  - Svantaggi: Necessità di meccanismo di sync

  L'area di memoria viene allocata da uno dei due processi. Entrambi si mettono d'accordo per sorpassare il limite dell'S.O. di non far scrivere nella mem. dell'altro.

MODELLO PRODUCER/CONSUMER

È un paradigma che descrive gli elementi comunicanti come:

- Producer: Produce i dati da inviare, ha un ruolo di potere.
- Consumer: Consuma i dati in arrivo, subisce i dati del producer.

Le informazioni possono essere scambiate mediante un buffer che può essere:

- Bounded: Il buffer ha dimensione prefissata. → La coda può riempirsi
- Un-bounded: Il buffer non ha dimensione definita. La coda non può riempirsi.

IPC A MESSAGE PASSING > 3.6

Il modello implementa 2 operazioni fondamentali di send(A, message) e receive(B, message).

I messaggi possono avere lunghezza

- Fissa: Realizz. semplice. Creare app.: più complessa
- Variabile: // . più complessa. // : semplice.

CANALE DI COMUNICAZIONE 1 (3.6.1)

Può essere a comunicazione

- Diretta: Viene specificato il destinatario
- Indiretta: Ha 1 prop (processo (SO)) → non può morire
  → se muore manca mail
  Si invia il messaggio in una casella / porta.
  Utile se si scrivono a 1+ interlocutori perché possono puntare alla casella.
  Il chiamante non ha info sulla ricezione

CANALE DI COMUNICAZIONE (3.6.1)

Il canale può anche essere classificato come:

- **Blocking / Synchronous**

  Il mittente e/o il destinatario rimangono in attesa dell'altro:

  -> Blocking send: Il mittente aspetta che il messaggio venga ricevuto

  -> Blocking receive: Il destinatario aspetta il messaggio

- **Non Blocking / Asynchronous**

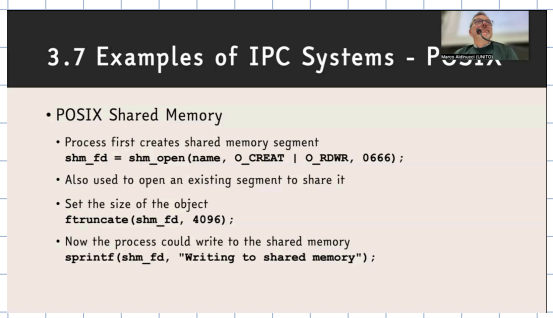
  Il mittente e/o il destinatario non attendono

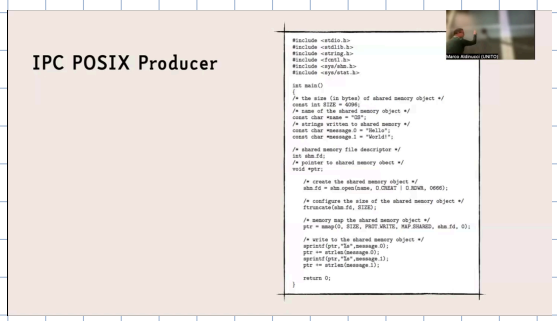
  -> N.B. send: Il chiamante prosegue dopo l'invio

  -> NB. Receive: Il dest. riceverà il msg o NULL e andrà avanti.

I messaggi si scambiano con una coda di messaggi che può essere dimensione 0 (sender aspetta), finita di N elementi (sender manda finché c'è spazio), infinita (limitata dalla memoria)

Mem. cond. con POSIX > 3.7





mmap consente di sincronizzare l'area di mem condivisa tra gli elem. comunicanti.
