

# I/O asincrono

Il programma richiede l'inizio di un'operazione di I/O, gli viene immediatamente restituito il controllo, e richiede in un secondo momento il risultato (tramite una chiamata o interrupt). È possibile effettuare più operazione asincrone contemporaneamente (*multiplexing*), per esempio tramite **select**: si specifica una lista di fd, e il programma viene bloccato finché uno qualsiasi non è disponibile per la lettura/scrittura (o dopo un timeout configurabile).

Idea alla base dell'*event-driven programming*: ciclo infinito, ad ogni iterazione si gestisce l'evento di I/O successivo.

È un approccio alternativo al creare un thread per ogni operazione di I/O. Spesso si combinano le due strategie: un thread per core, e ciascuno fa multiplexing su I/O tramite **select** o simili.