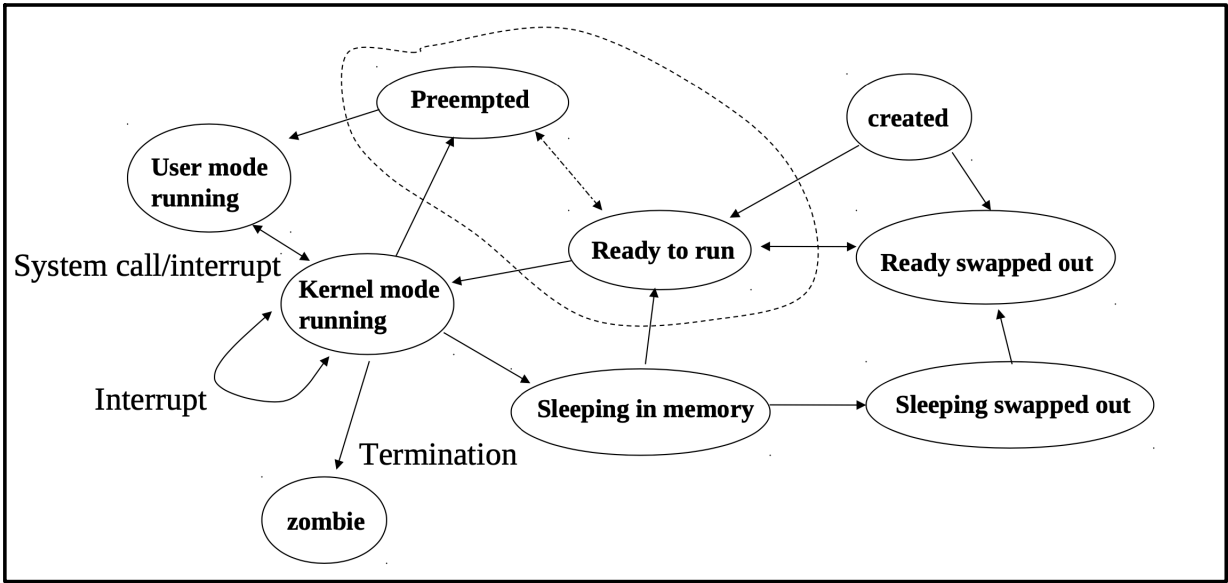


Reference UNIX state diagram



- La parte running viene suddivisa in due stati: noi possiamo essere in "running user mode" e "running kernel mode".  
Noi possiamo arrivare in modalità kernel anche a causa di un interrupt: siamo sempre noi, abbiamo solamente una variazione di flusso d'esecuzione.
- Però quando il software del sistema ha il controllo può decidere di assegnare la CPU a qualcun altro: in quel caso noi andiamo a finire in uno stato denominato "PREEMPTED" e di conseguenza possiamo essere portati a "ready to run".
- Il kernel può portarci anche in stato "SLEEP", magari siamo in attesa che il dispositivo di I/O completi la sua esecuzione. E nel frattempo il software del Kernel cede la CPU a qualcun altro e noi rimaniamo in "sleeping in memory".
- Ovviamente il software del Kernel può anche decidere di portare il nostro Address Space fuori dalla ram con un'operazione di "Swap Out".

Se chiamo una System Call per uscire, chiamo il software del Kernel e quest'ultimo ci porta in uno stato ZOMBIE. Perché questa applicazione è ancora presente nel sistema, non riprenderà più il controllo della CPU, però persistiamo all'interno di questo stato zombie. Il sistema di controllo per un po' terrà traccia degli zombie.  
Se noi abbiamo un blocco di codice che va a chiamare la system\_call EXIT(0), e stiamo indicando al kernel che questa applicazione non vuole più avere il controllo della CPU, ma non solo, IL CODICE 0 CHE VIENE PASSATO, ESSO SI CHIAMA CODICE DI TERMINAZIONE!!!

Nel PCB dell'applicazione che è in stato ZOMBIE, verrà mantenuto il codice di terminazione che abbiamo comunicato, che può essere 0,1,2 etc..!  
Quindi il PCB dell'applicazione verrà comunque mantenuto dal software del sistema.

Classica immagine di processo in sistemi UNIX		
Testo	Contesto utente	
Dati		
Stack utente		
Memoria condivisa		
Program counter	Contesto registri	
Registro di stato del processore		
Stack pointer		
Registri generali		
Entry nella tabella dei processi	Contesto sistema	
U area (area utente)		
Tabella indirizzamento (memoria virtuale)		
Stack del modo kernel		

Entry della tabella dei processi - campi principali

- Stato del processo
- Identificatori d'utente (reale ed effettivo)

Informazioni relative all'utenza

Identificatori d'utente (reale ed effettivo)

Associata ad un programma attivo.

- Identificatori di processi (pid, id del genitore)
- Descrittore degli eventi (valido in stato sleeping)
- Affinità di processore (insieme di processori utili per l'esecuzione)
- Priorità 

Più core di CPU  
I core scritti dentro l'affinità sono quelli che possono gestire il processo!
- Segnali (mandati al processo ma non ancora gestiti)
- Timer (monitoring)
- Stato della memoria (swap in/out)

I segnali sono informazioni che riguardano "come dovrà avvenire la gestione all'interno di questo processo, di segnalazioni che possono provenire dall'esterno del processo"

### U area - campi principali

- Identificatori d'utente (effettivo/reale)
- Array per i gestori di segnali
- Terminale
- Parametri di I/O (es. indirizzi dei buffer)
- Timer (monitoring in modalità utente)
- Valore di ritorno di system calls
- Tabella dei descrittori di file