



Computação Paralela: Processos, Threads e Sincronização e Thread Queues

[Processos](#)

[Threads](#)

[Estados](#)

[Controlo de Execução](#)

[Java](#)

Processos

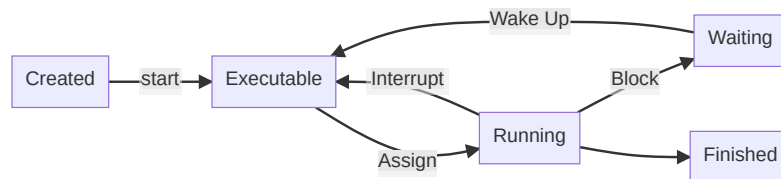
- Um processo pode ser considerado como a execução de um programa:
 - Contêm toda a informação necessária para a sua execução: dados, *stack*, registos e instruções a serem executadas;
 - Cada processo tem o seu próprio espaço de endereçamento com acesso exclusivo (i.e. nenhum outro processo pode aceder a esta memória).

Threads

- A criação de *threads* é geralmente mais rápida que a criação de processos;
- As *threads* podem ser fornecidas pelo *runtime* (***user-level threads***) ou pelo SO (***kernel-level threads***):
 - *User-Level Threads*:
 - Correm em qualquer SO;
 - Menos privilégios *low-level*;
 - Não permite que aplicações *multithreaded* corram diferentes *threads* em diferentes processadores;
 - *Kernel-Level Threads*:

- Gestão ocorre no kernel;
- Permite que o mesmo processo seja escalonado em diferentes *kernel-level threads*;
- Criação e gestão difícil.

Estados



- **Created:** a *thread* foi criada mas ainda não executou nenhuma operação;
- **Executable:** a *thread* está pronta para ser executada, não estando ainda associada a algum recurso;
- **Running:** a *thread* está a correr;
- **Waiting:** a *thread* está à espera de um evento externo (i.e. I/O);
- **Finished:** A *thread* terminou a sua execução.

Controlo de Execução

- **Sincronização por barreira:** define um ponto em que cada *thread* terá que esperar pela execução das restantes;
- **Sincronização por condição:** uma *thread* T1 é bloqueada até que uma certa condição seja cumprida.

Java

- Estender a classe `Thread`:
 - Mais flexível (permite a utilização de mais métodos);
 - Cada *thread* é uma instância nova da classe que estendeu `Thread`;
- Implementar a interface `Runnable`:
 - Possui apenas o método `run`;
 - Útil quando a classe em questão já estende outra classe qualquer.

Mecanismos de Sincronização

- **Race conditions:**

- Acontecem quando o sistema tenta realizar 2 operações ao mesmo tempo;
- Poderão levar a estados indesejados num *software*.