



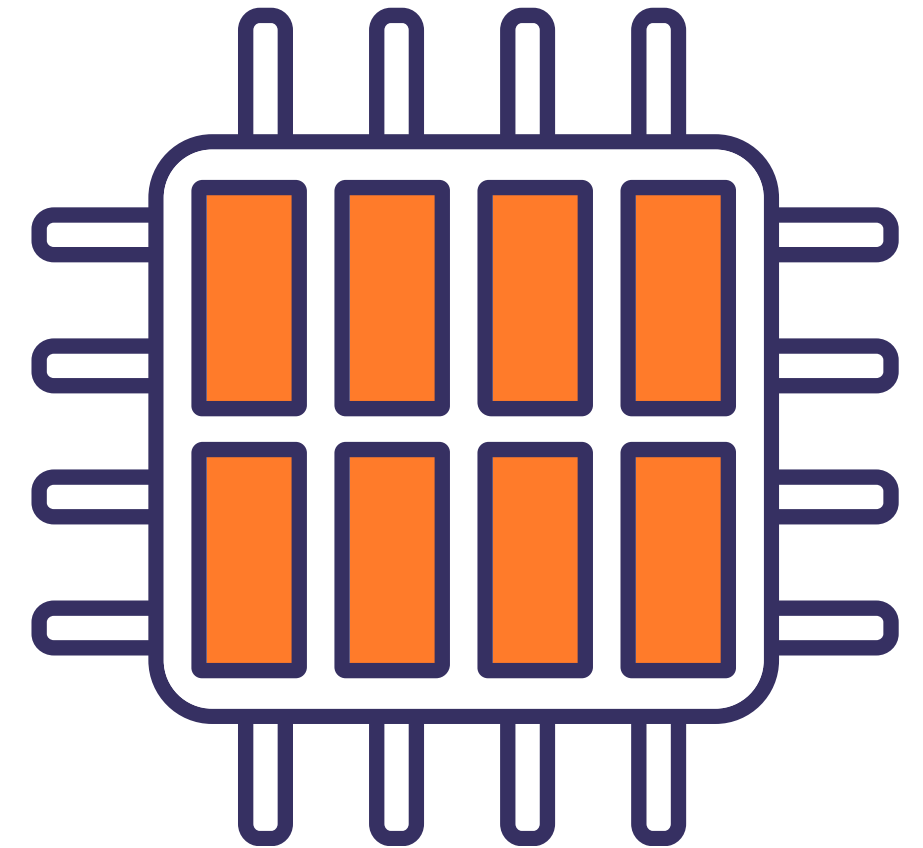
SD – Intro

Modèles de parallélisme

Programmation parallèle

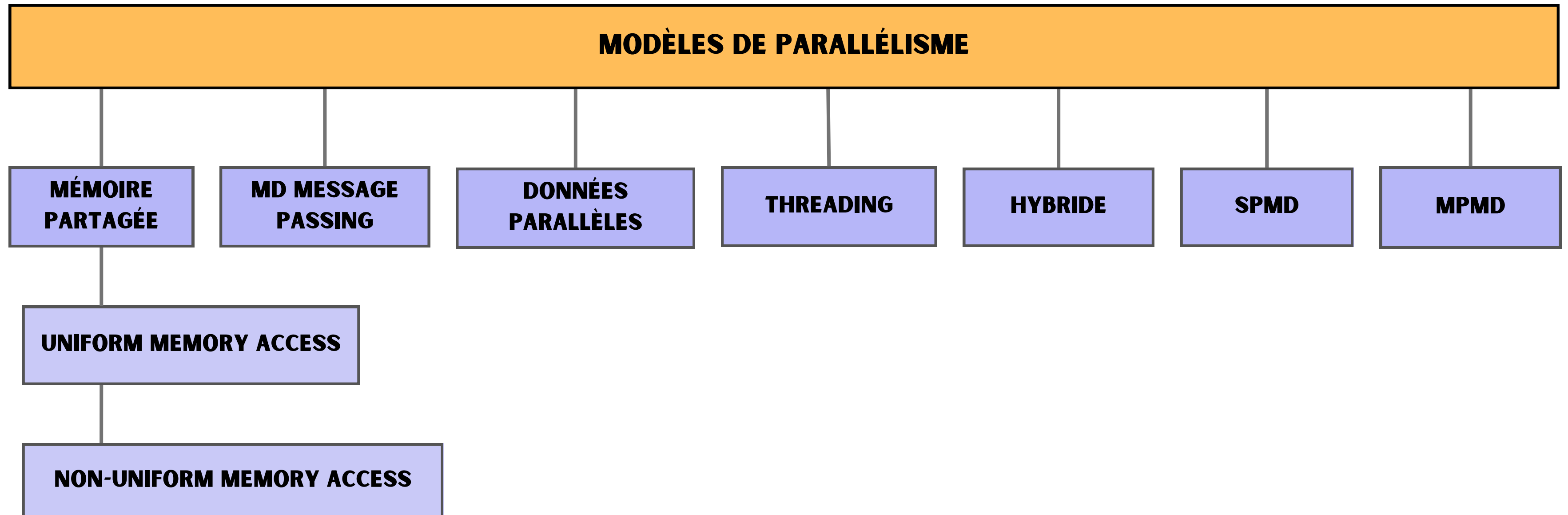
Distributed systems

Last update: Fév. 2024



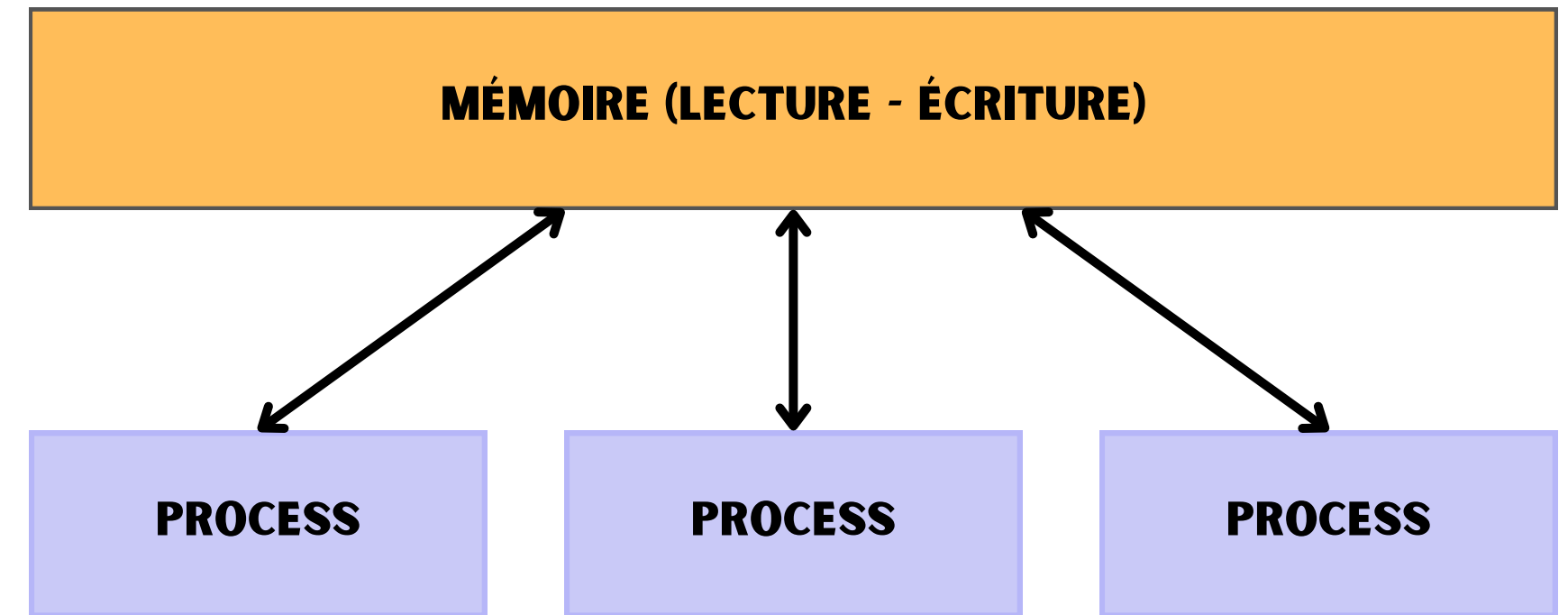
MODÈLES DE PARALLÉLISME

VUE GLOBALE SUR LES MODÈLES DE PROGRAMMATION PARALLÈLE



MODÈLES DE MÉMOIRE PARTAGÉE

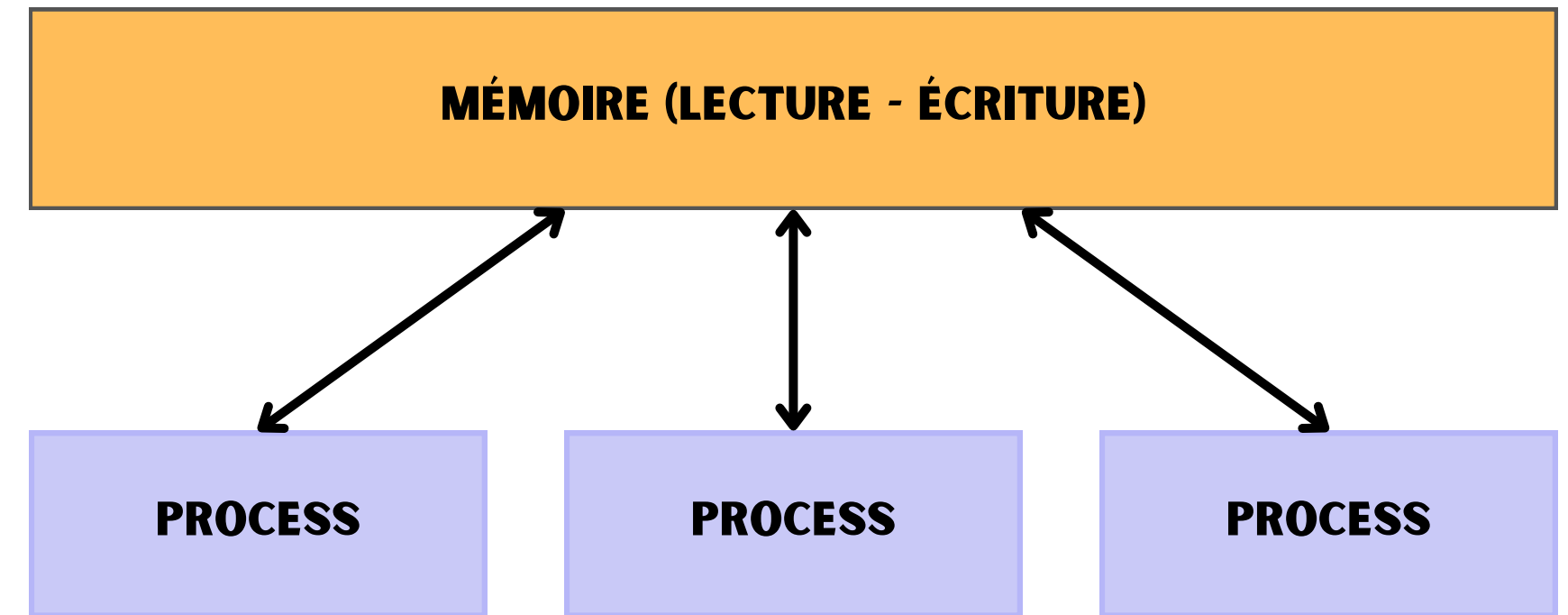
- Dans ce modèle de programmation, les processus/tâches partagent un espace d'adressage commun, qu'ils lisent et écrivent de manière asynchrone.
- Divers mécanismes tels que les verrous et les sémaphores sont utilisés pour contrôler l'accès à la mémoire partagée, résoudre les conflits et prévenir les conditions de course et les blocages.



MODÈLES DE MÉMOIRE PARTAGÉE

Exemple d'implémentation

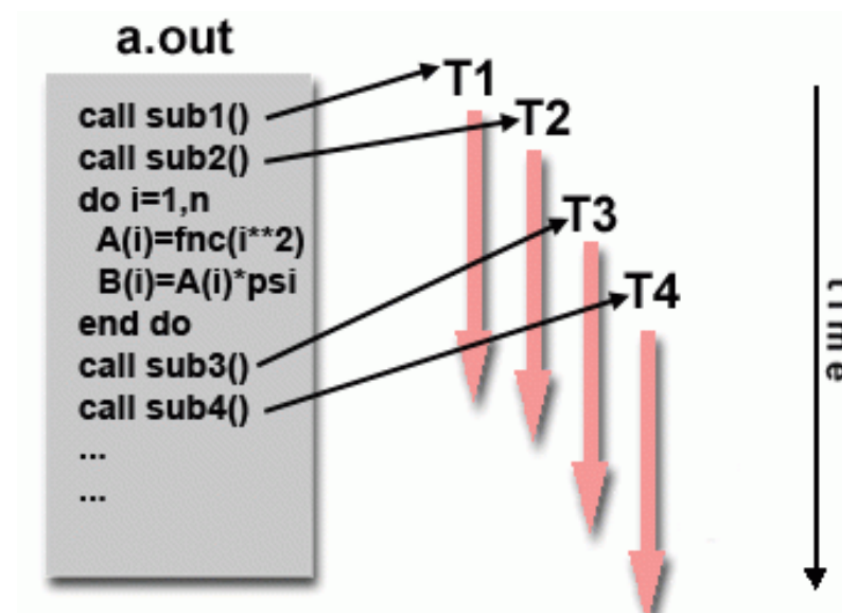
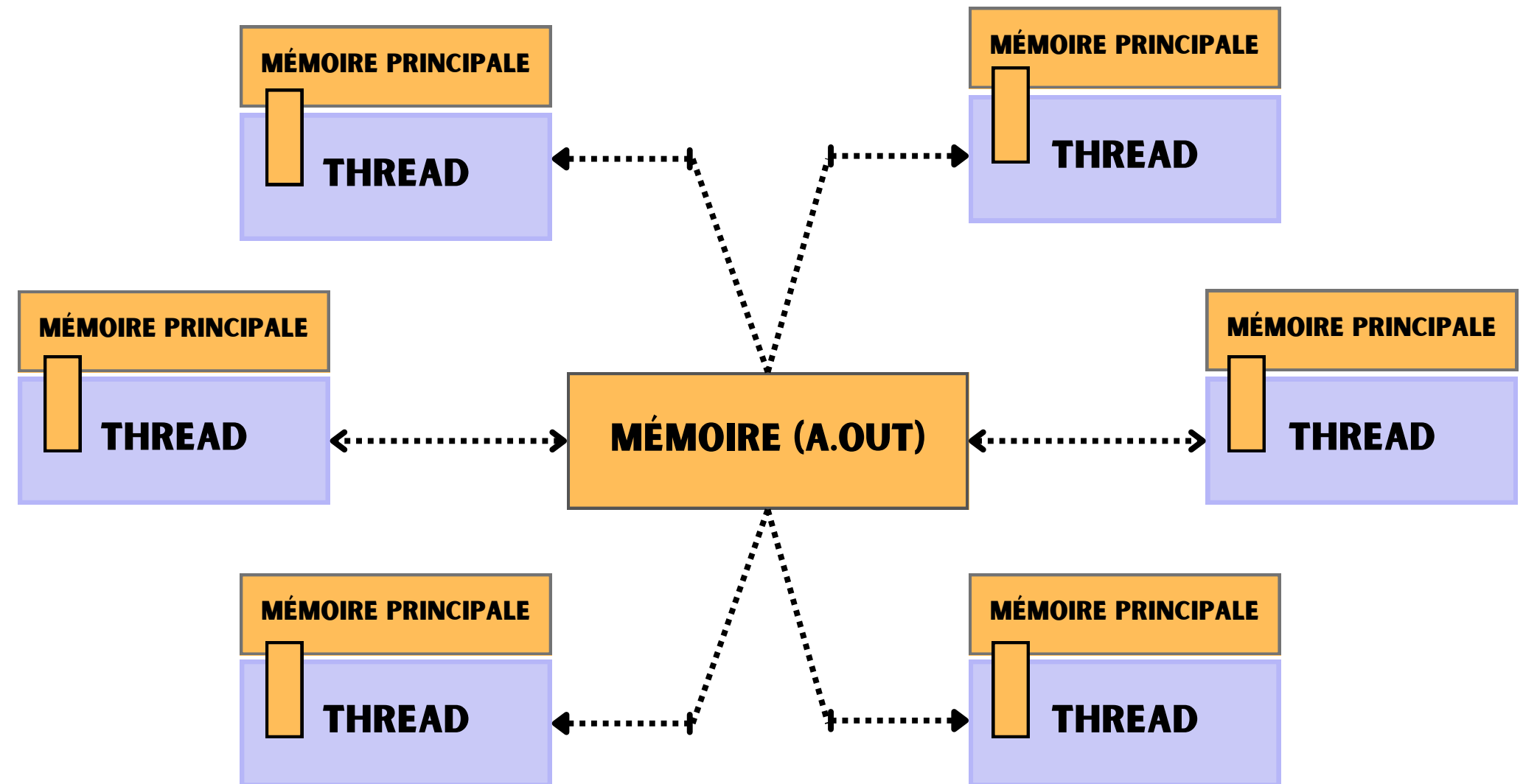
- Sur les machines autonomes à mémoire partagée, les systèmes d'exploitation natifs, les compilateurs et/ou le matériel fournissent un support pour la programmation en mémoire partagée
- Norme POSIX fournit une API pour l'utilisation de la mémoire partagée, et UNIX fournit des segments de mémoire partagée (shmget, shmat, shmctl, etc.)



THREADING

Dans le modèle **THREAD**, un seul processus "lourd" peut avoir plusieurs voies d'exécution concurrentes (threads) "légères".

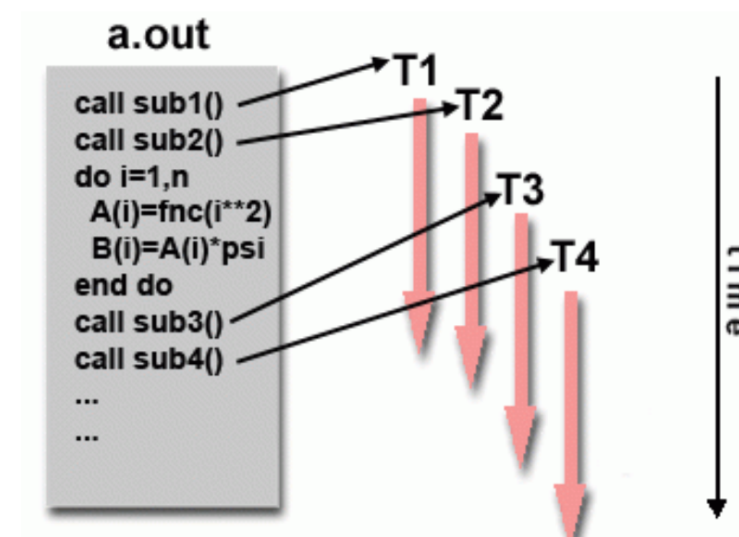
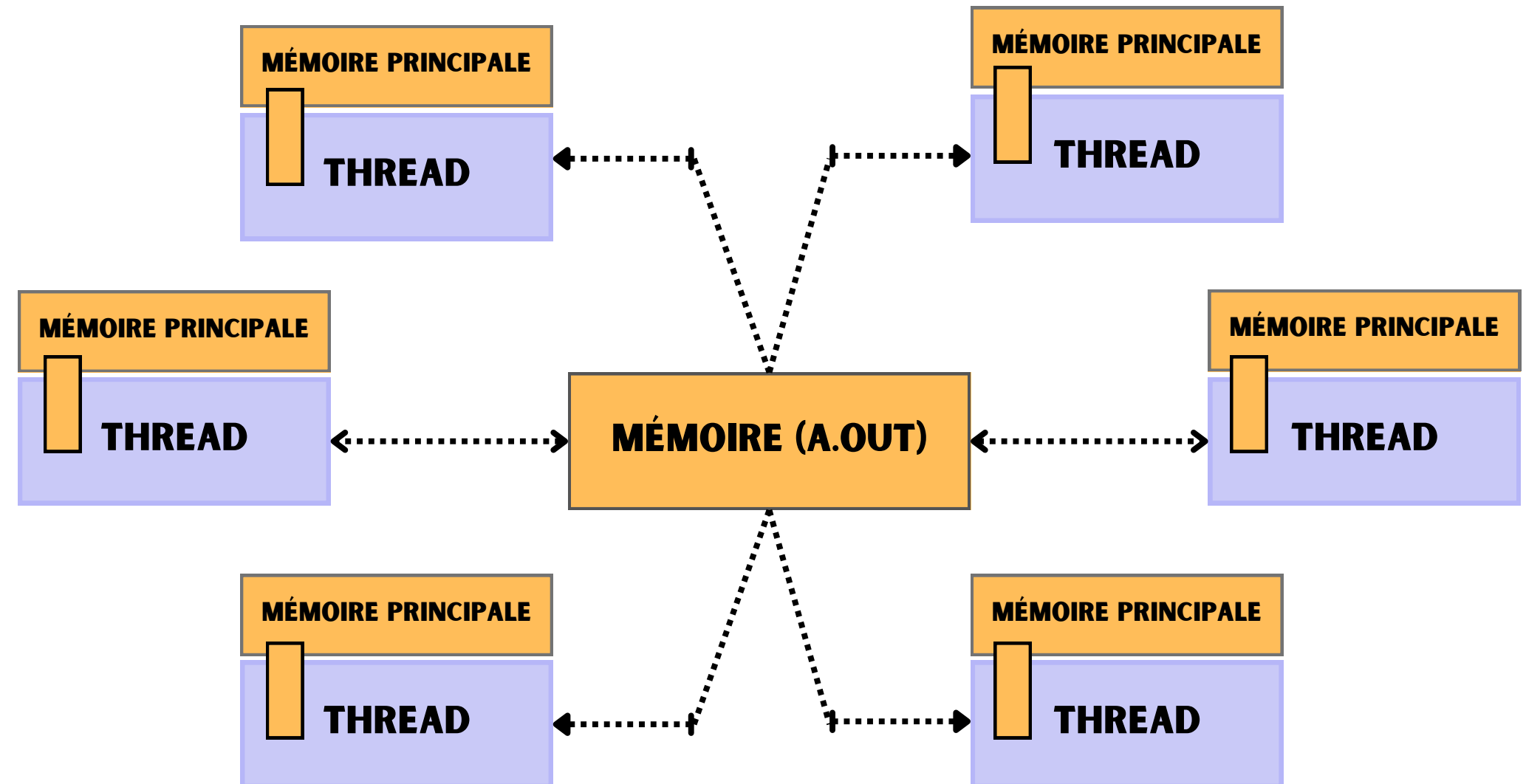
- PAR EXEMPLE, DANS CETTE FIGURE NOUS AVONS 6 THREAD POUR UN PROGRAMME A.OUT.
- LES THREADS SE PARTAGEANT LA MÉMOIRE GLOBALE DU PROCESSUS (A.OUT), ET AVEC CHACUN POSSÈDE SA MÉMOIRE LOCALE.



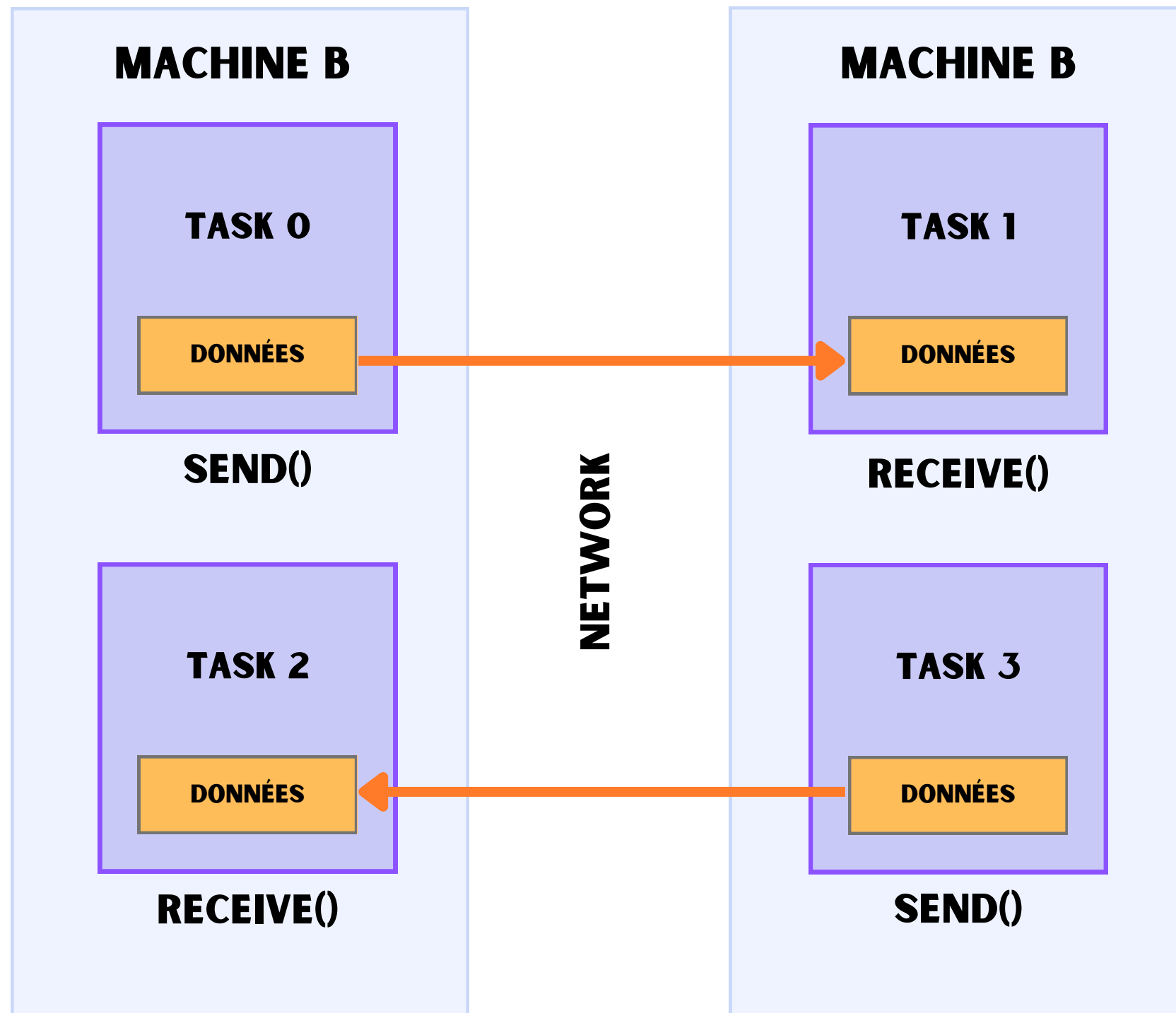
THREADING

Exemple d'implémentation

- POSIX Threads
- OpenMP
- Microsoft threads
- Java, Python threads
- CUDA threads pour GPUs

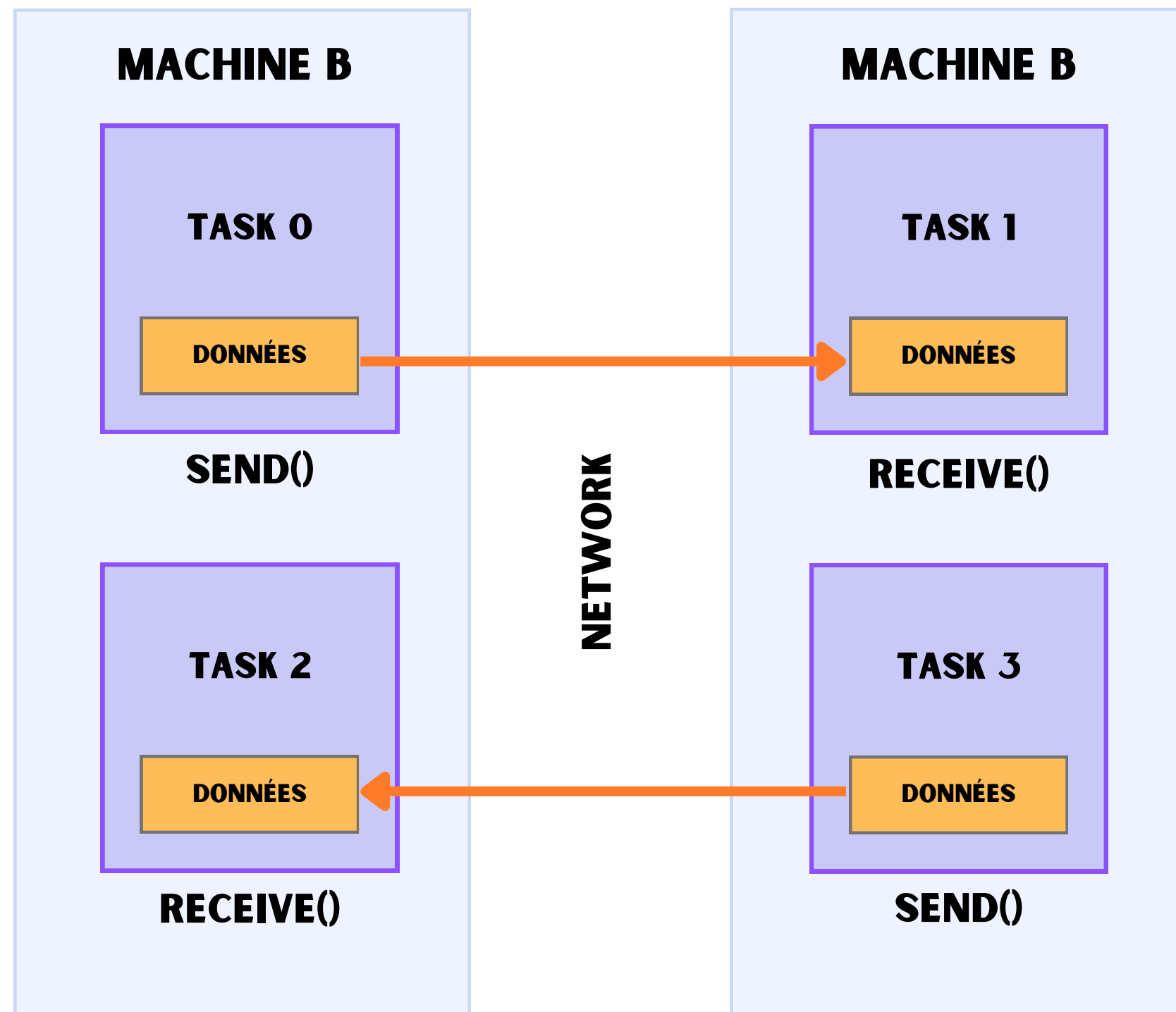


LE MODÈLE MÉMOIRE DISTRIBUÉE-PASSAGE DE MESSAGE



- Un ensemble de tâches utilisent leur propre mémoire locale pendant le calcul.
- Plusieurs tâches peuvent résider sur la même machine physique et/ou sur un nombre arbitraire de machines.
- Échange de données par le biais de communications en envoyant et en recevant des messages.
- Transfert de données nécessitant des opérations coopératives de réception correspondante (rendez-vous).
- MPI est un exemple de ce modèle.

LE MODÈLE MÉMOIRE DISTRIBUÉE-PASSAGE DE MESSAGE

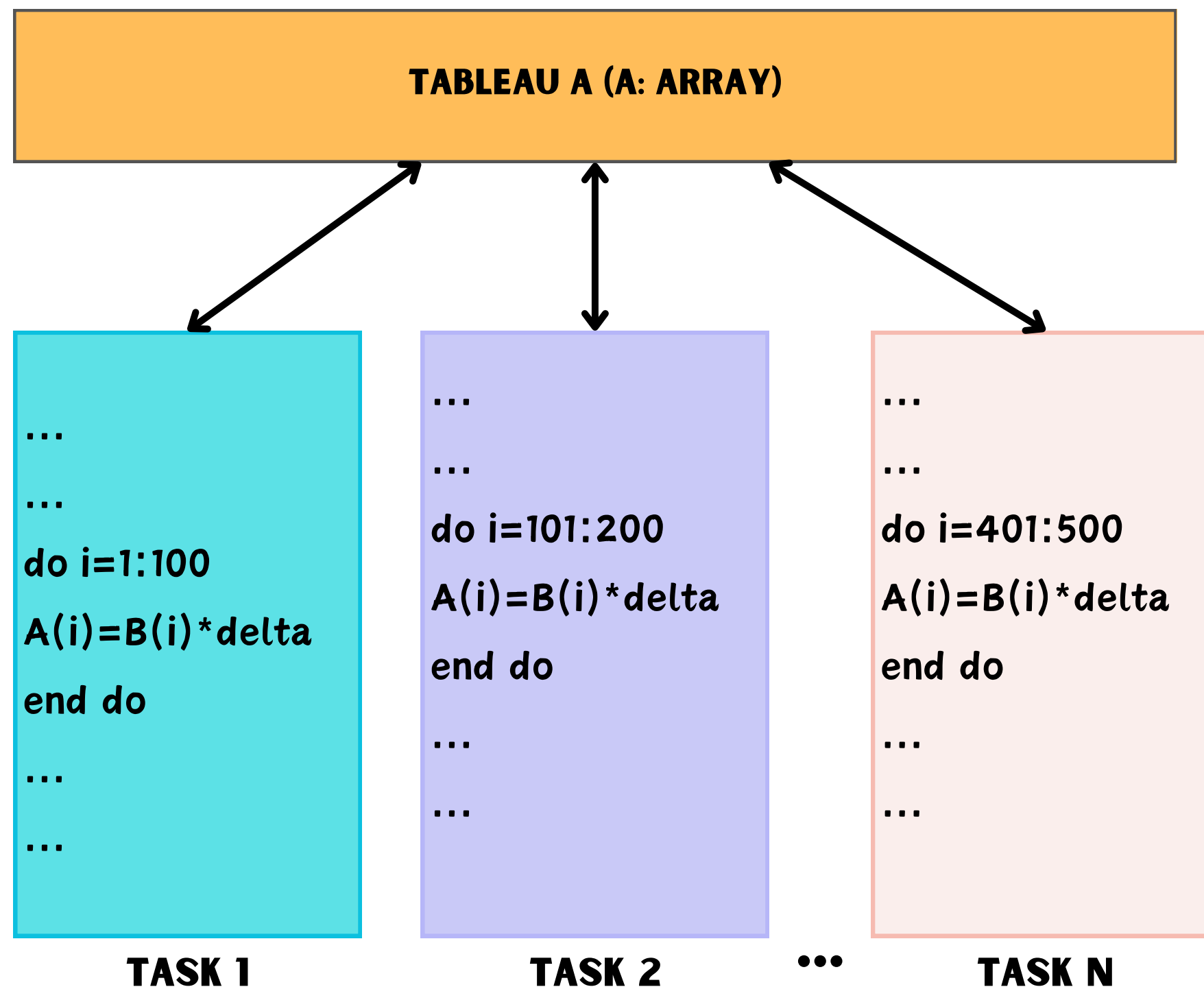


Exemple d'implémentation

- **MPI** est un exemple de ce modèle

MODÈLE DE DONNÉES PARALLÈLES

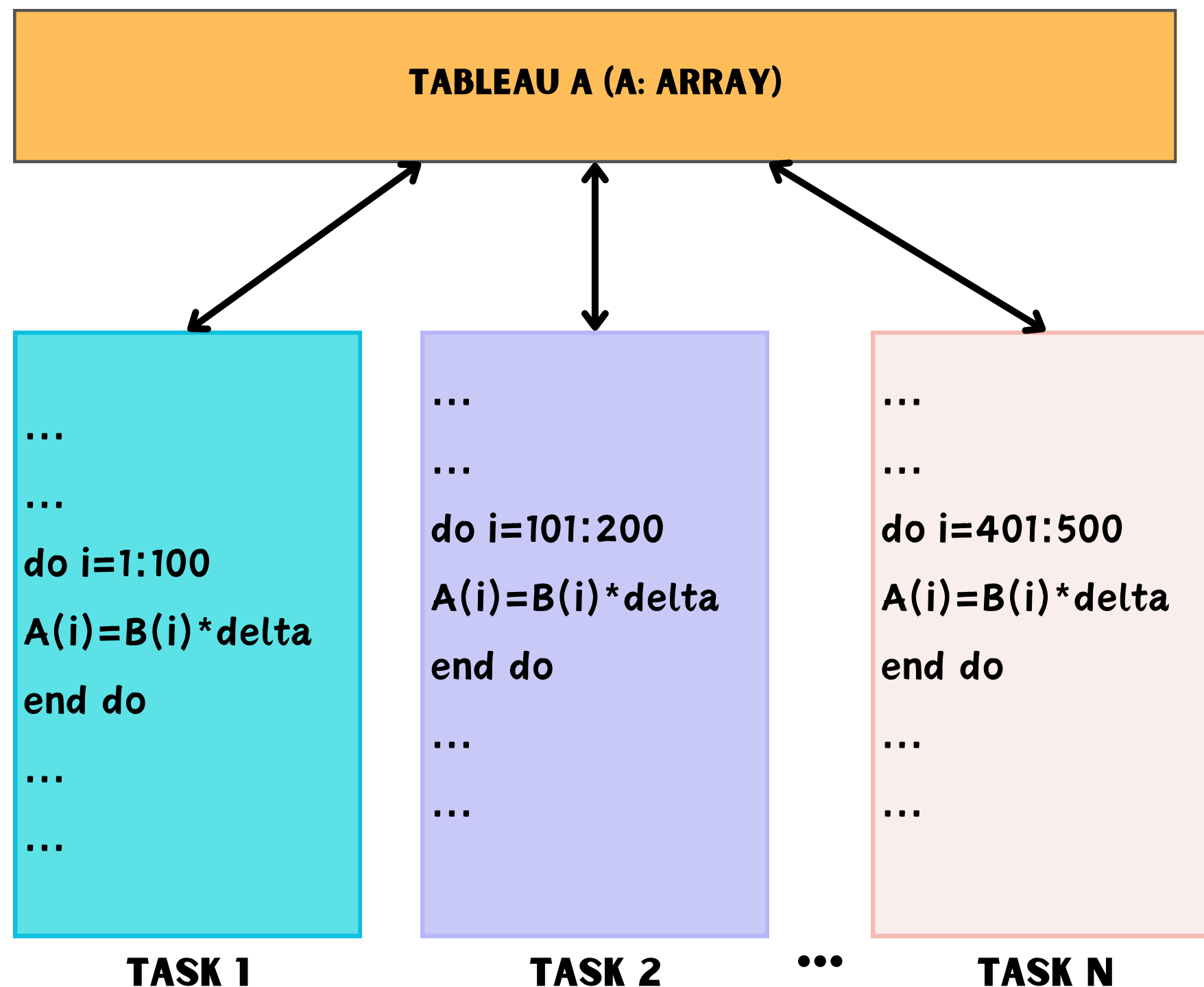
**TRAITEMENT PARALLÈLE D'UN TABLEAU DE 1000 ÉLÉMENTS
DIVISION EN N TÂCHES, AVEC CHACUNE 100 ÉLÉMENTS À TRAITER**



- Espace d'adressage traité globalement
- Ensemble de tâches travaillant collectivement sur la même structure de données, avec chaque tâche opérant sur une partition différente de la même structure de données
- Les tâches effectuent la même opération sur leur partition de travail

MODÈLE DE DONNÉES PARALLÈLES

TRAITEMENT PARALLÈLE D'UN TABLEAU DE 1000 ÉLÉMENTS
DIVISION EN N TÂCHES, AVEC CHACUNE 100 ÉLÉMENTS À TRAITER

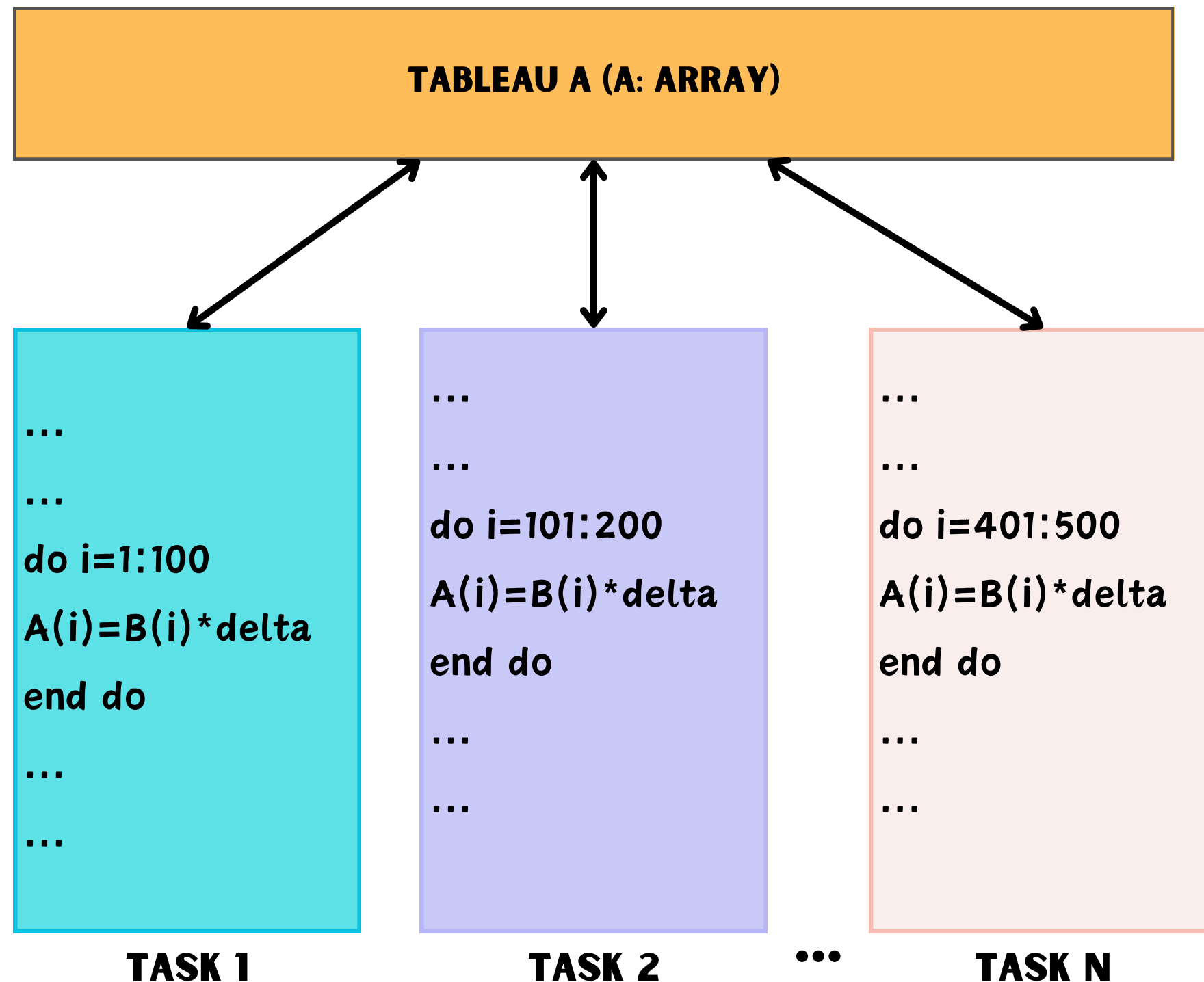


Sur les architectures à mémoire partagée, toutes les tâches peuvent avoir accès à la structure de données via la mémoire globale.

Sur les architectures à mémoire distribuée, la structure de données globale peut être divisée logiquement et/ou physiquement entre les tâches.

MODÈLE DE DONNÉES PARALLÈLES

**TRAITEMENT PARALLÈLE D'UN TABLEAU DE 1000 ÉLÉMENTS
DIVISION EN N TÂCHES, AVEC CHACUNE 100 ÉLÉMENTS À TRAITER**

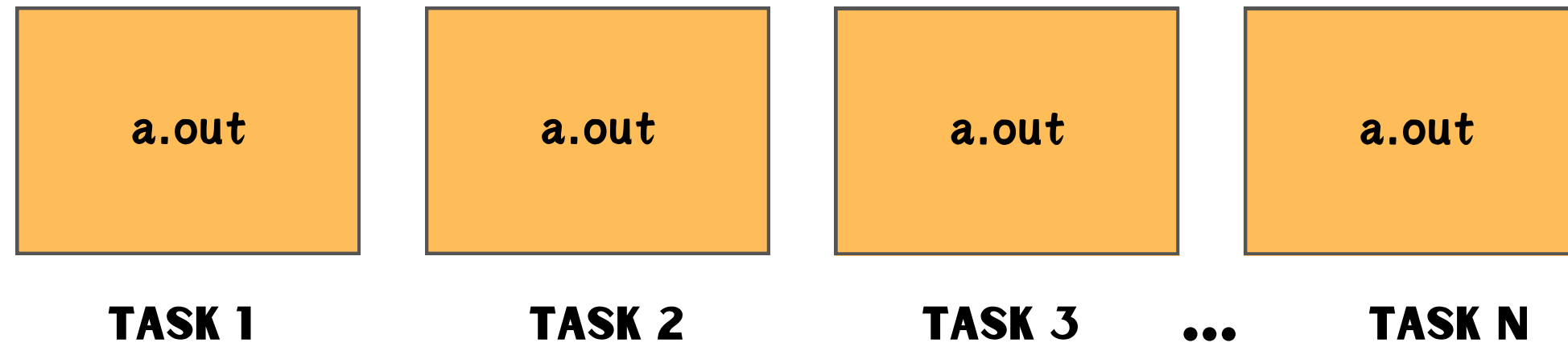


Exemples d'implémentation

- **Coarray Fortran**
- **Unified Parallel C (UPC)**
- **Global Arrays**
- **X10**
- **Chapel**

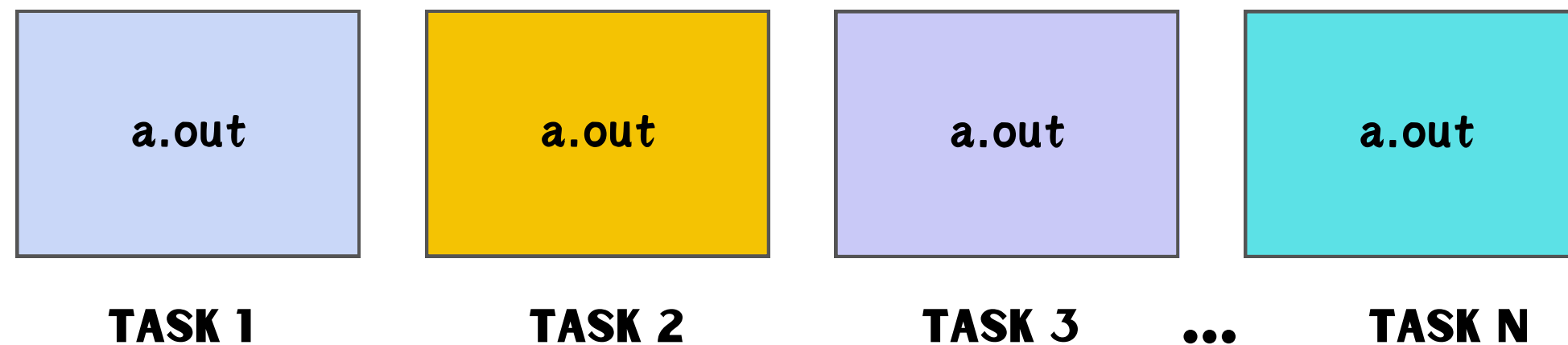
SPMD - MPMD

SPMD : **Single** Program Multiple Data



SPMD est un modèle de programmation de "haut niveau" qui peut être construit à partir de n'importe quelle combinaison des modèles de programmation parallèle mentionnés précédemment.

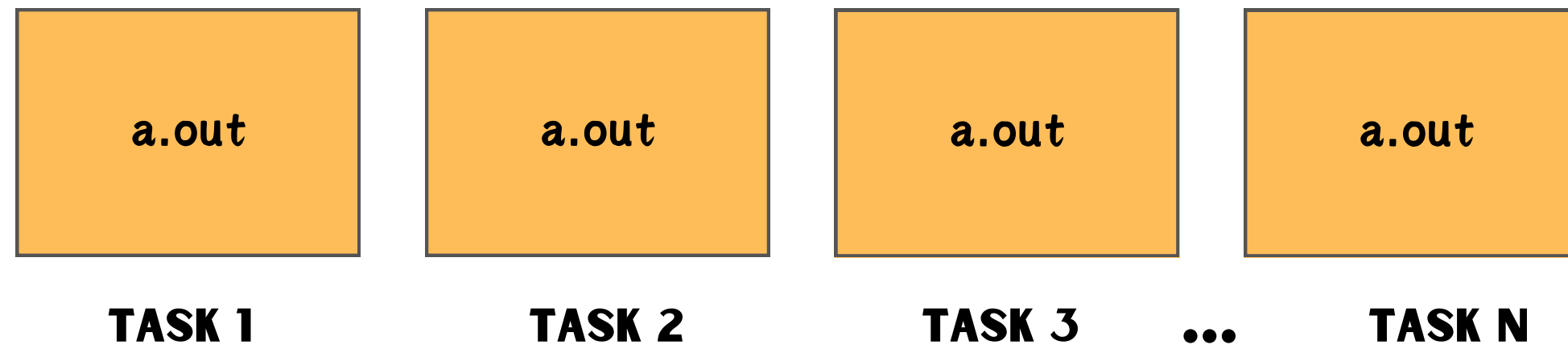
MPMD : **Multiple** Program Multiple Data



MPMD est en fait un modèle de programmation de "haut niveau" qui peut être construit à partir de n'importe quelle combinaison des modèles de programmation parallèle mentionnés précédemment.

SPMD - MPMD

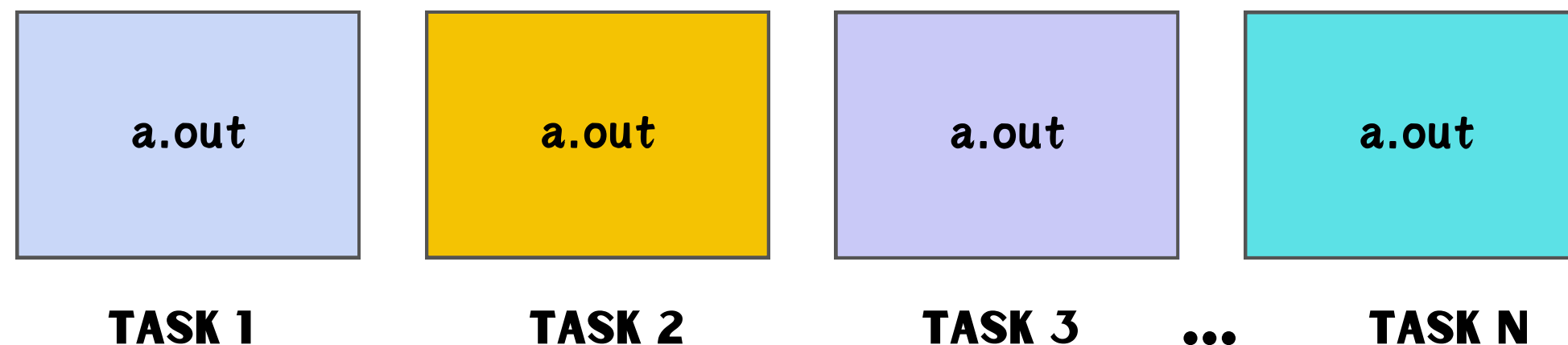
SPMD : **Single** Program Multiple Data



PARTITION DE DONNÉES

- Les tâches peuvent exécuter différents programmes simultanément.
- Les programmes peuvent être de type "threads", "message passing", "data parallel" ou hybride.

MPMD : **Multiple** Program Multiple Data

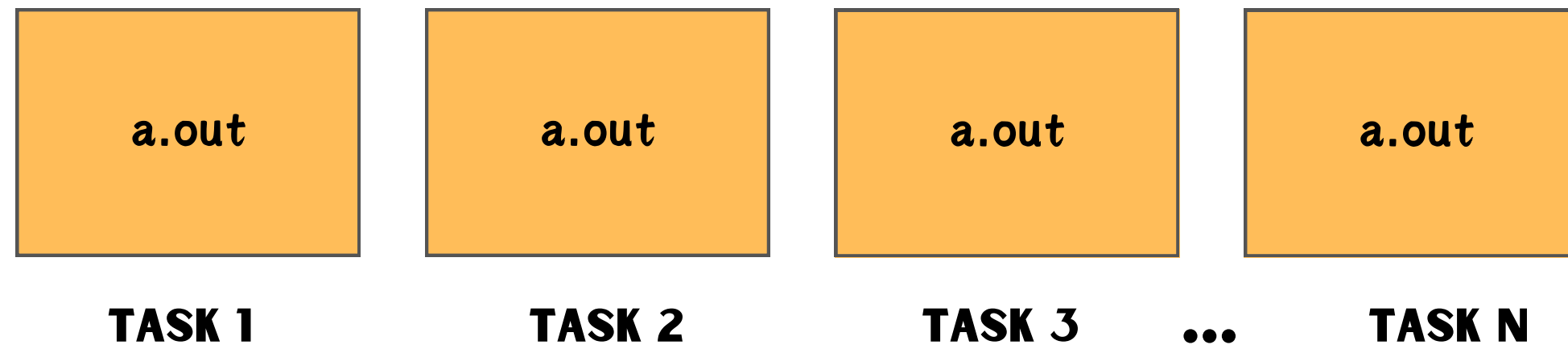


PARTITION FONCTIONNEL

- Toutes les tâches exécutent simultanément leur copie du même programme.
- Ce programme peut être de type "threads", "message passing", "data parallel" ou hybride.

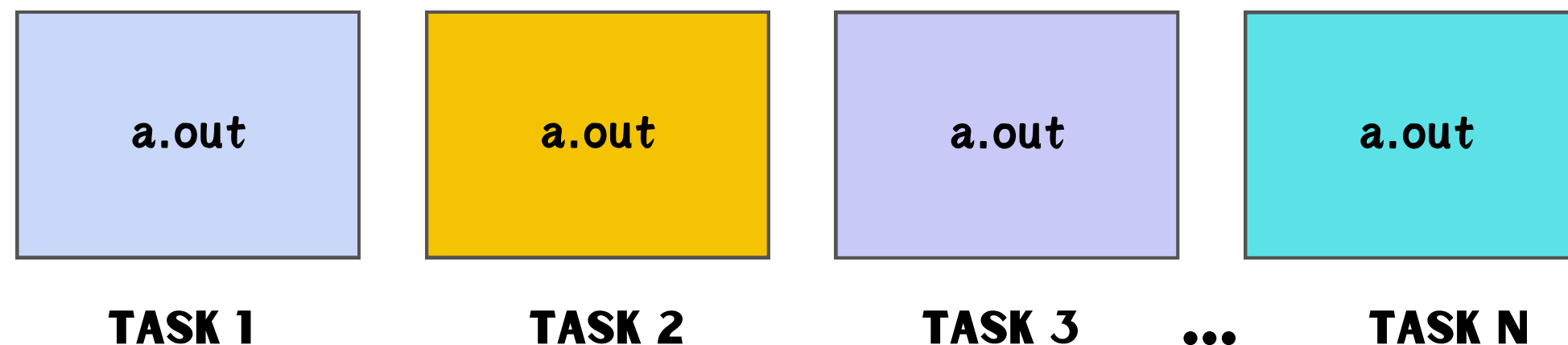
SPMD - MPMD

SPMD : **Single** Program Multiple Data



- Les tâches peuvent exécuter différents programmes simultanément.
- Les programmes peuvent être de type "threads", "message passing", "data parallel" ou hybride.

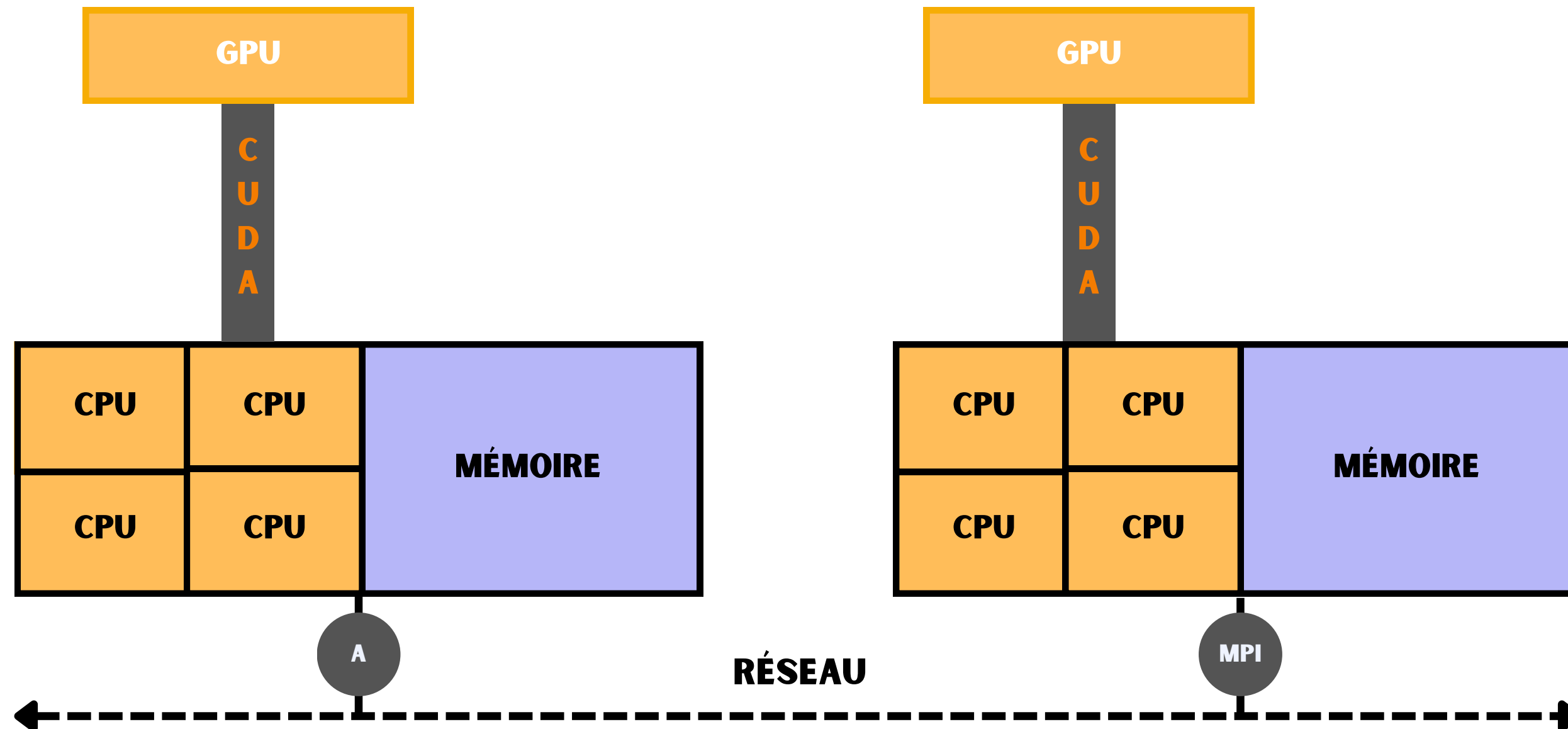
MPMD : **Multiple** Program Multiple Data



- Toutes les tâches exécutent simultanément leur copie du même programme.
- Ce programme peut être de type "threads", "message passing", "data parallel" ou hybride.

AUTRES MODÈLES : HYBRIDE

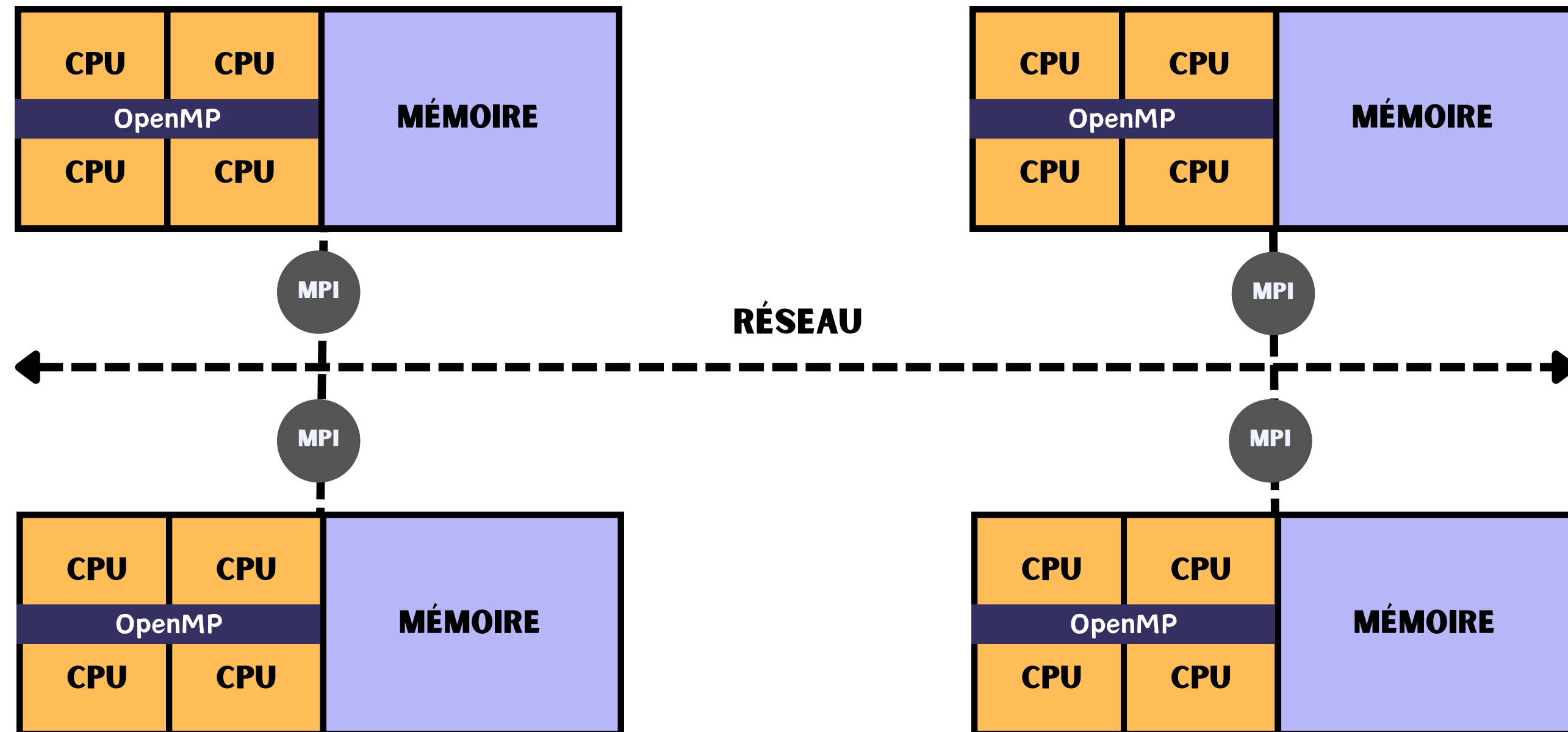
Les modèles hybrides sont des combinaisons de plusieurs modèles simples



MODÈLE HYBRIDE MPI - CUDA

AUTRES MODÈLES : HYBRIDE

Les modèles hybrides sont des combinaisons de plusieurs modèles simples



MODÈLE HYBRIDE MPI - OPENMPI