Architecture des ordinateurs

61 - Gestion de la mémoire centrale
La mémoire virtuelle

Philippe Darche
IUT Paris Descartes

Rappel

- A l'origine
  - une petite mémoire primaire (quelques Kio)
  - pour un seul programme
  - pour un seul utilisateur

Trois problèmes

- Taille limitée de la mémoire physique
  - besoin d'un grand espace mémoire
  - ⇒ la réponse : le mécanisme de mémoire virtuelle
- Risque d'utilisation involontaire ou volontaire d'une ressource
  - ⇒ besoin d'une protection
    - au départ logicielle (prototypage du concept)
    - aujourd'hui matérielle (performance)
- Contexte de la multiprogrammation
  - plusieurs programmes en cours d'exécution sur un seul processeur
  - plusieurs utilisateurs connectés
  - ⇒ besoin d'une gestion fine de l'espace mémoire
    - déplacement par exemple

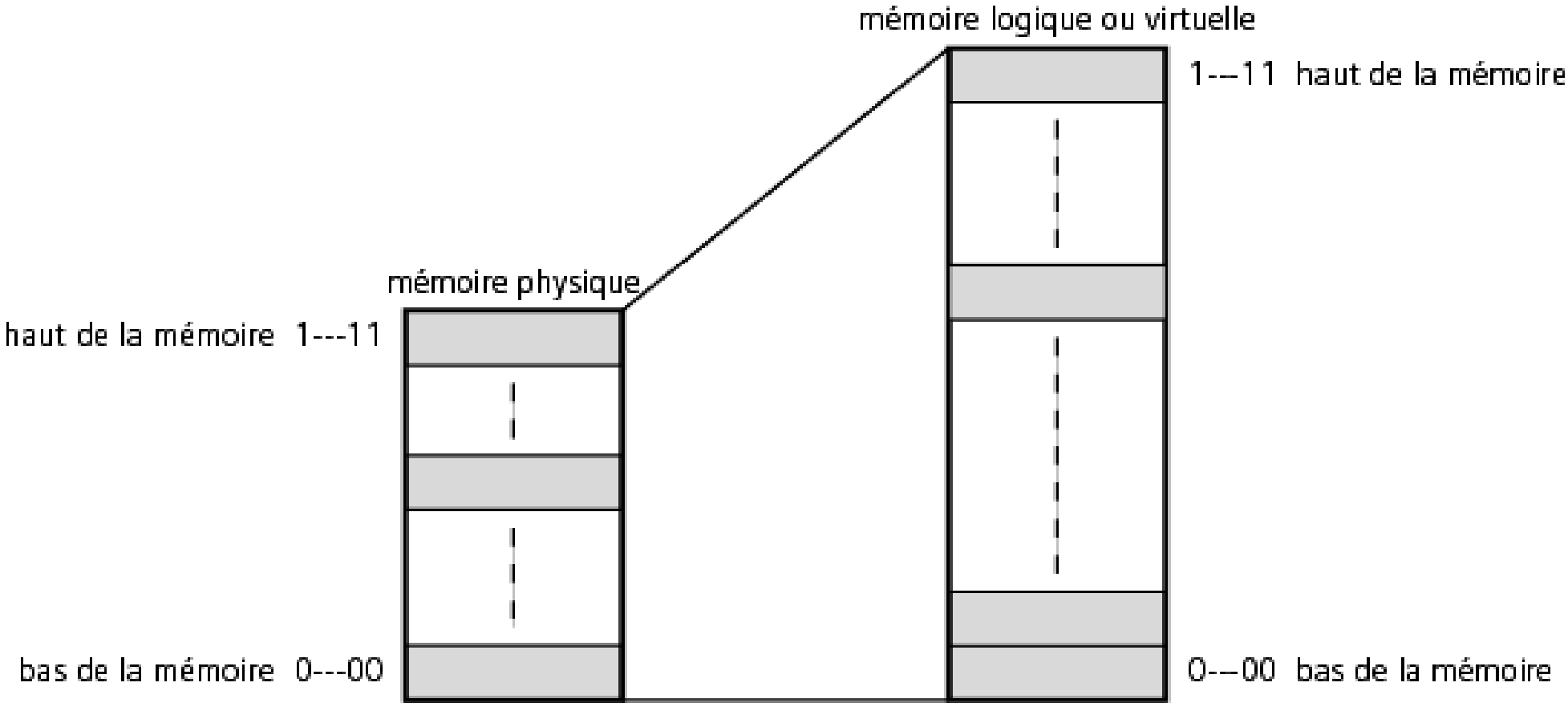
Gestion de la mémoire primaire

- Le cours ne concerne que la mémoire vive
  - le BIOS (ROM) n'est pas concerné
- Les mécanismes étudiés
  - la segmentation
  - la pagination

La mémoire virtuelle

- Zone mémoire logique de taille très supérieure à la Mémoire principale (MC)
  - ordre de grandeur du rapport : 2^{10}
- Repose sur l'utilisation d'une mémoire secondaire
  - l'unité de mémoire de masse à disques durs
- Taille limite : celle de la mémoire de masse
- Synonyme : mémoire logique

La mémoire virtuelle



Ordre de grandeur

- i286 :
  - 16 Mio de mémoire physique
  - 1 Gio de mémoire virtuelle
    - 16 384 segments de 64 Kio.
- i386 :
  - 4 Gio de mémoire physique (m = 32 bits)
  - 64 Tio de mémoire virtuelle
    - 16 384 segments de 4 Gio

Ordre de grandeur (suite)

- Pentium Pro
  - 64 Gio de mémoire physique (m = 36 bits)
  - 64 Tio de mémoire virtuelle (m’ = 46 bits)

- Microprocesseurs modernes (*i.e.* 64 bits)
  - 256 Gio de mémoire physique (m = 38 bits)
  - 256 Tio de mémoire virtuelle (m’ = 48 bits)

La compilation

- Traduction identificateur en adresse (logique)
  - variable
  - fonction (adresse de début)
- Exemple

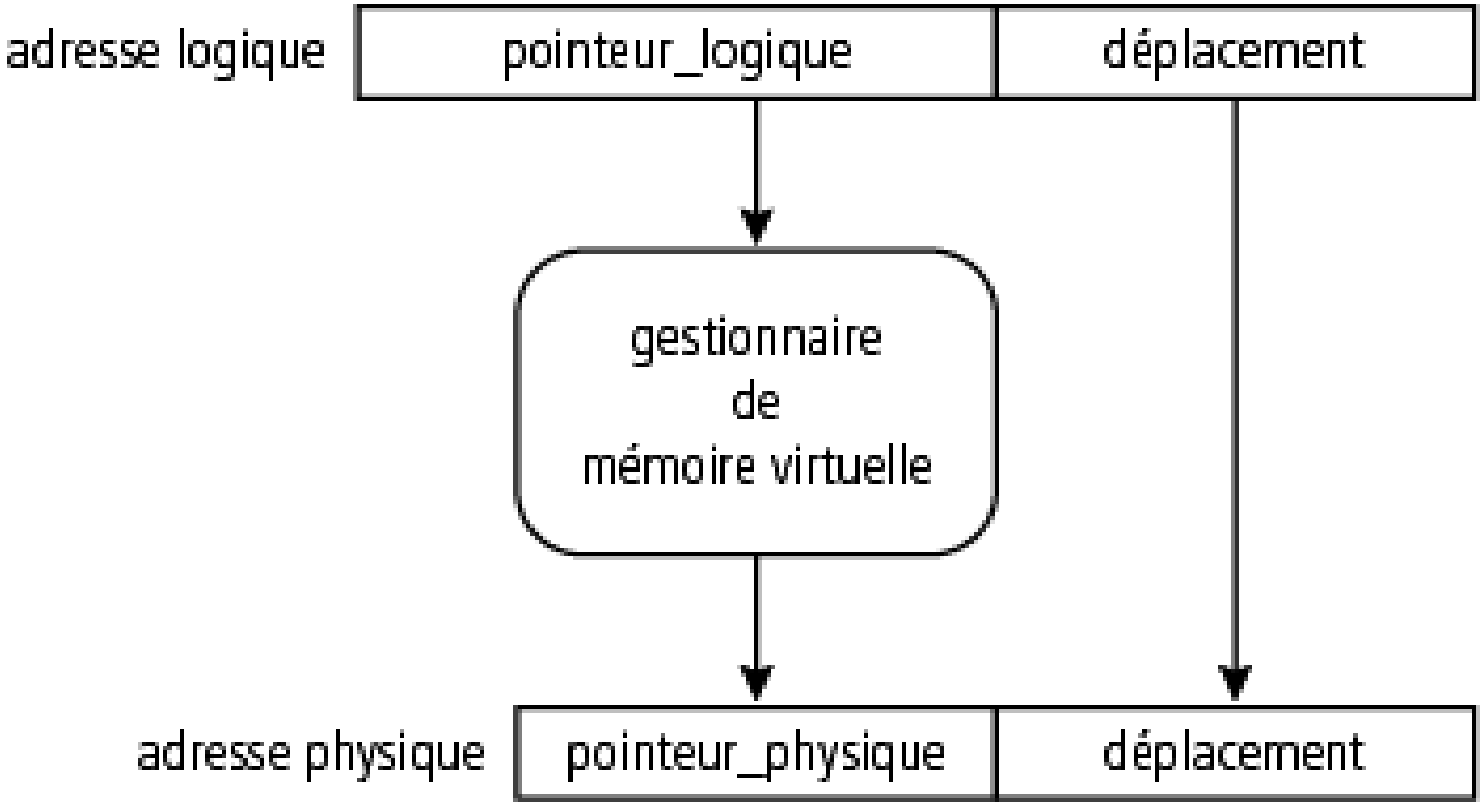
  A = A + 1 (langage C)

  équivalent en langage d'assemblage

```
mov AX,[A]
add AX,1
mov [A], AX
```

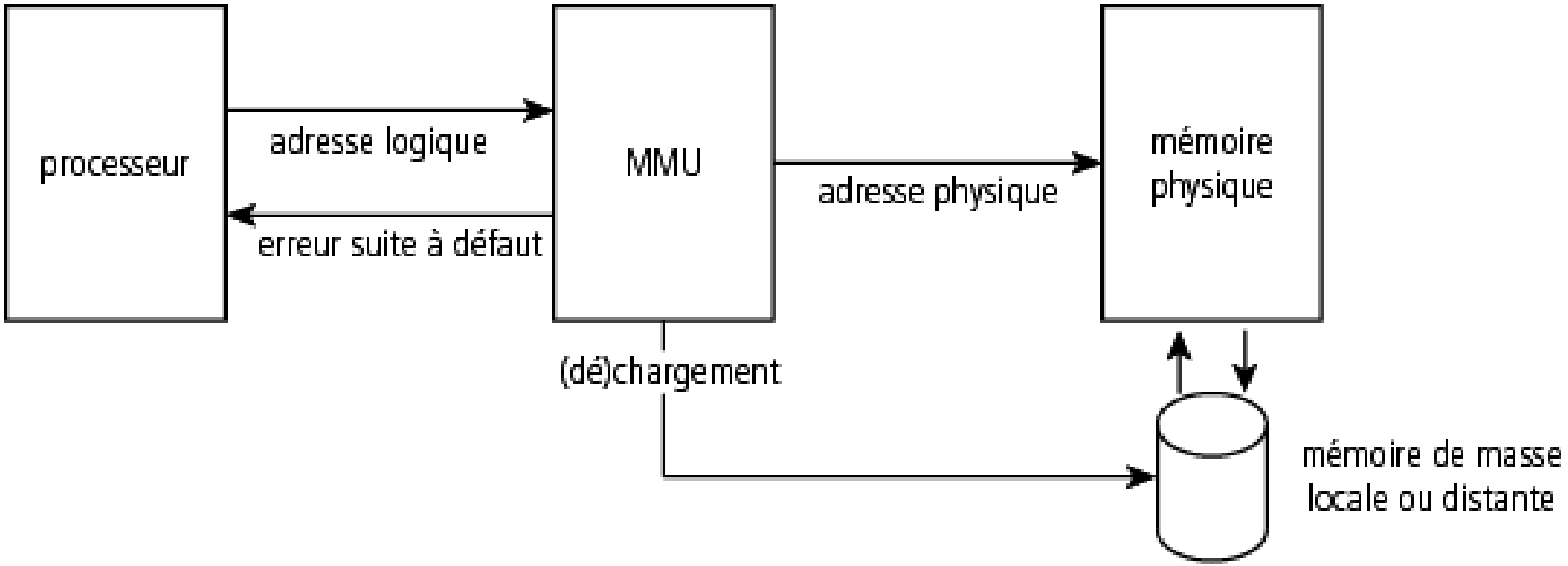
Transformation

- MMU : *Memory Management Unit*

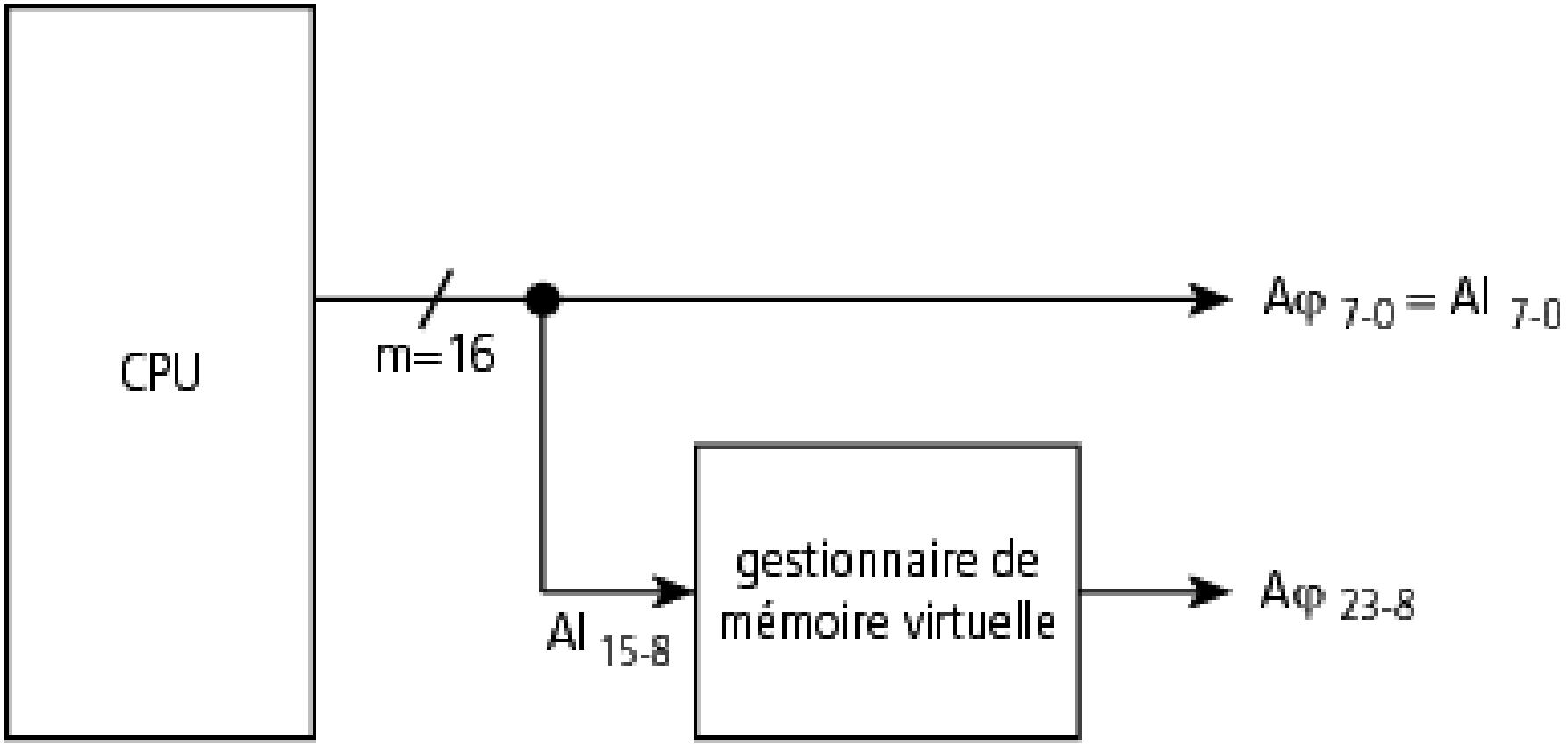


Transformation

- Adresse logique → adresse physique



Un chemin d'adresse

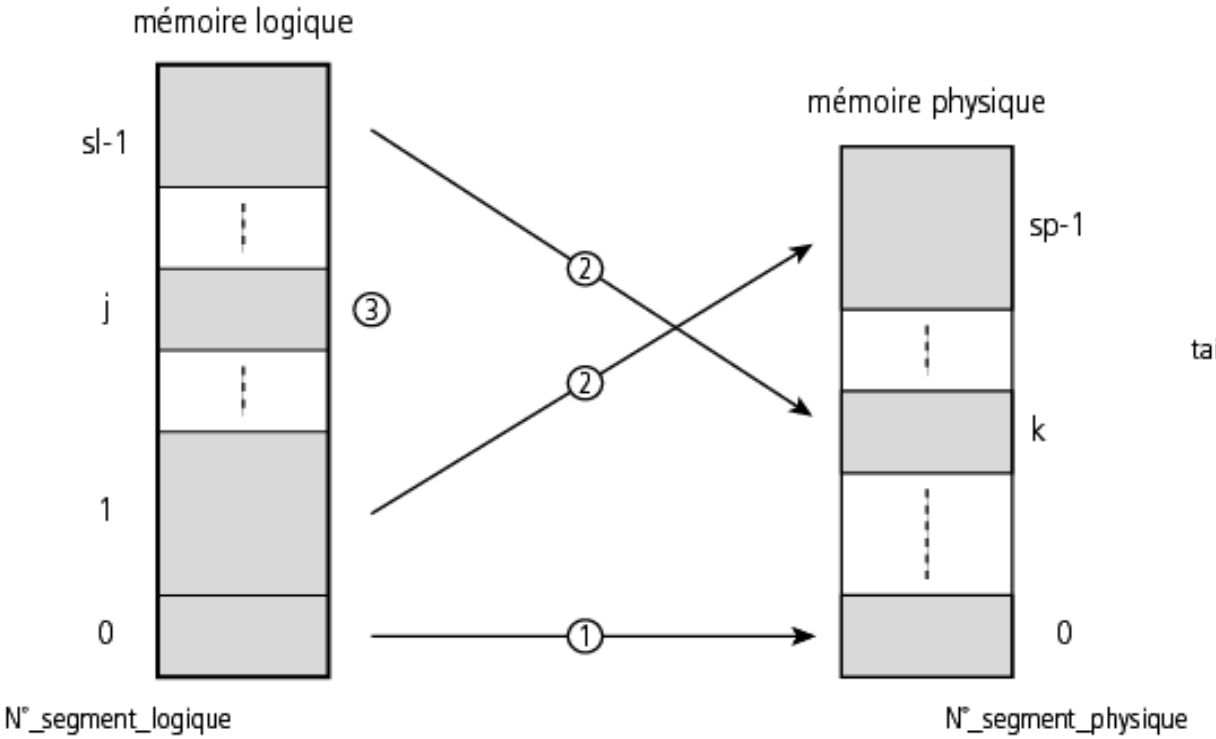


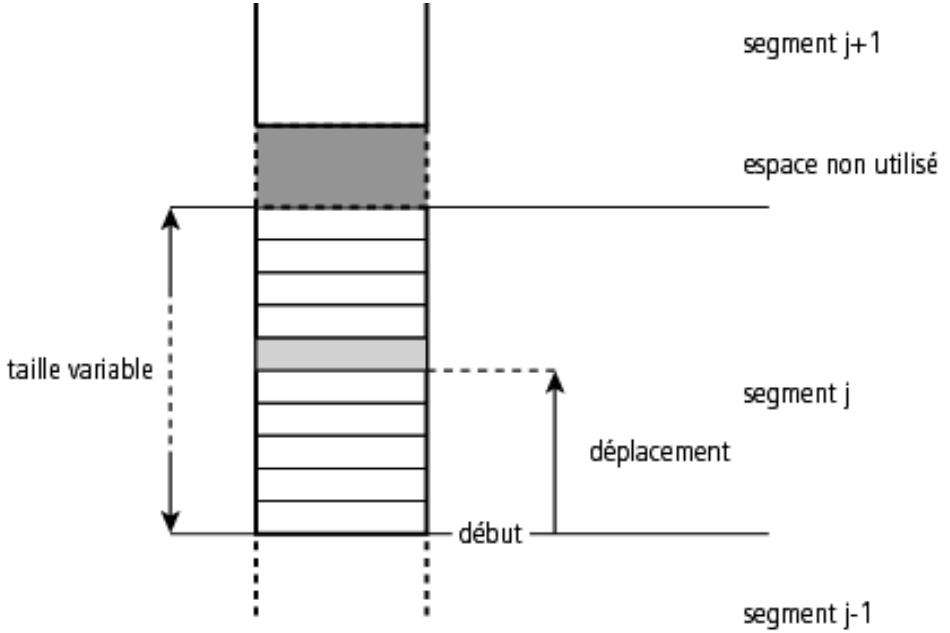
La segmentation

- Un segment
  - zone contigüe de mémoire
  - taille variable avec un maximum
    - i386 → 64 Kio
  - repéré en mémoire par un pointeur ad_début_seg
    - registre spécialisé
  - chaque mot est adressable par rapport au début du segment (*offset* ou décalage)
    - adressage relatif

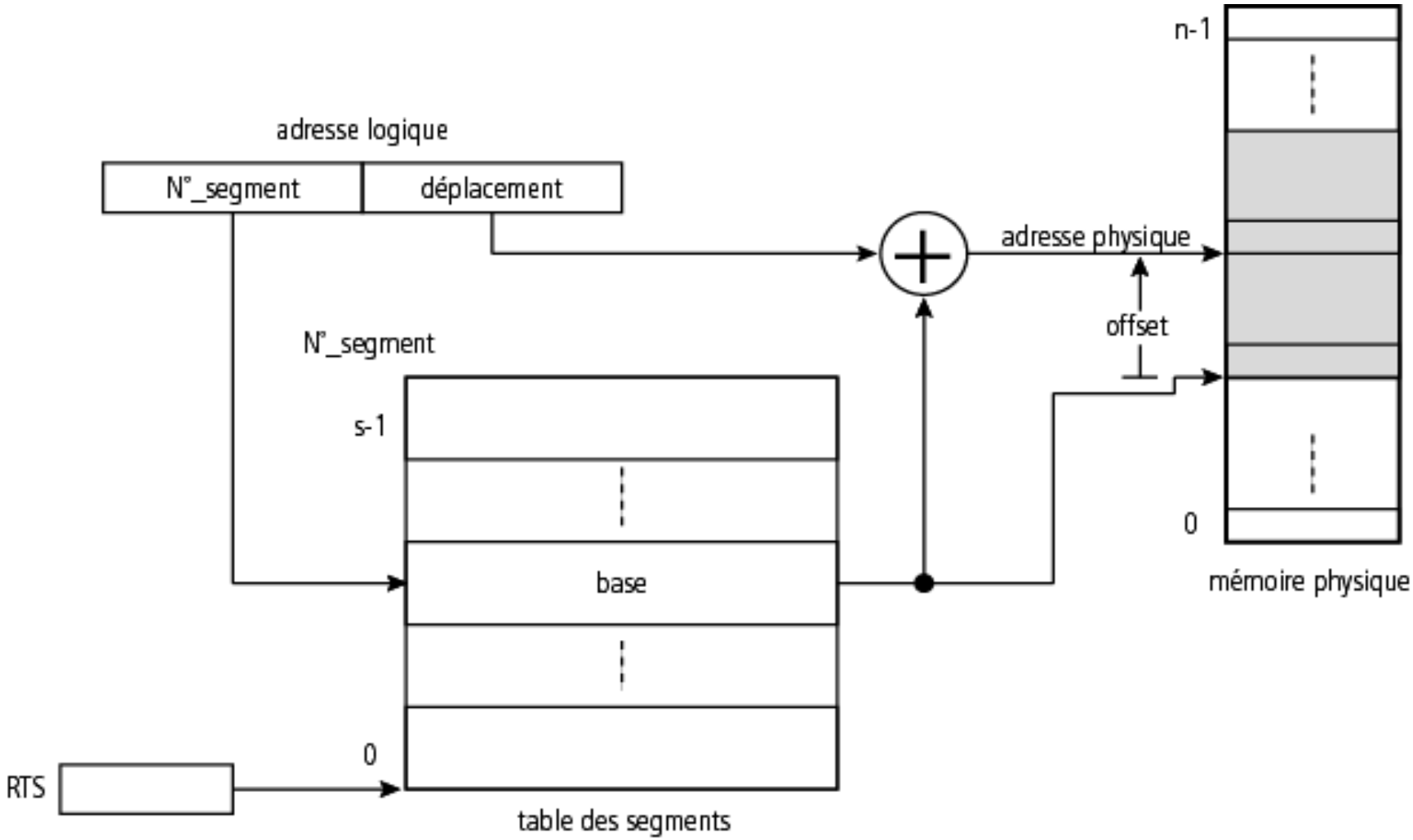
Les segments

- Segment en gris clair

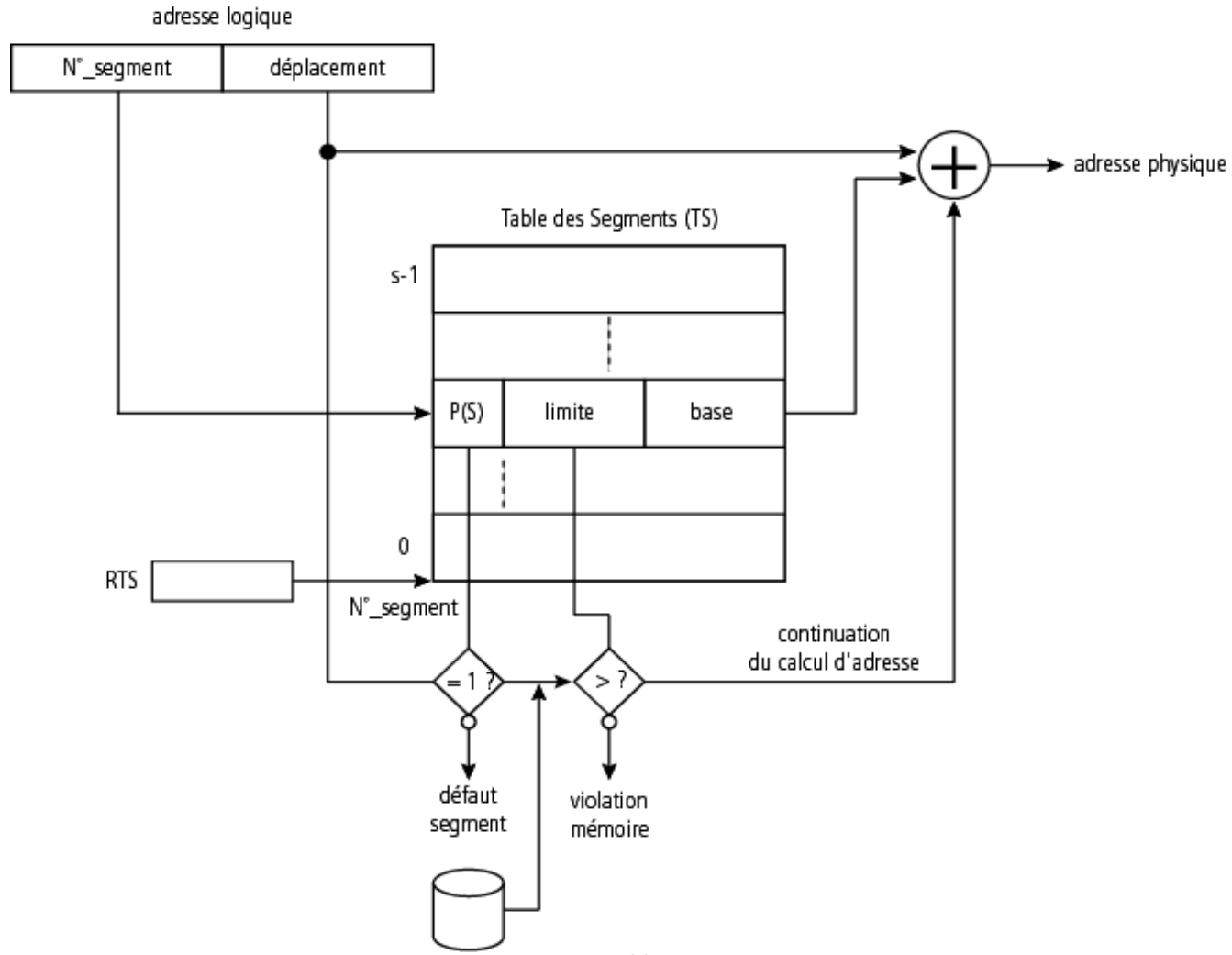




Calcul de l'adresse physique



Mécanismes de protection

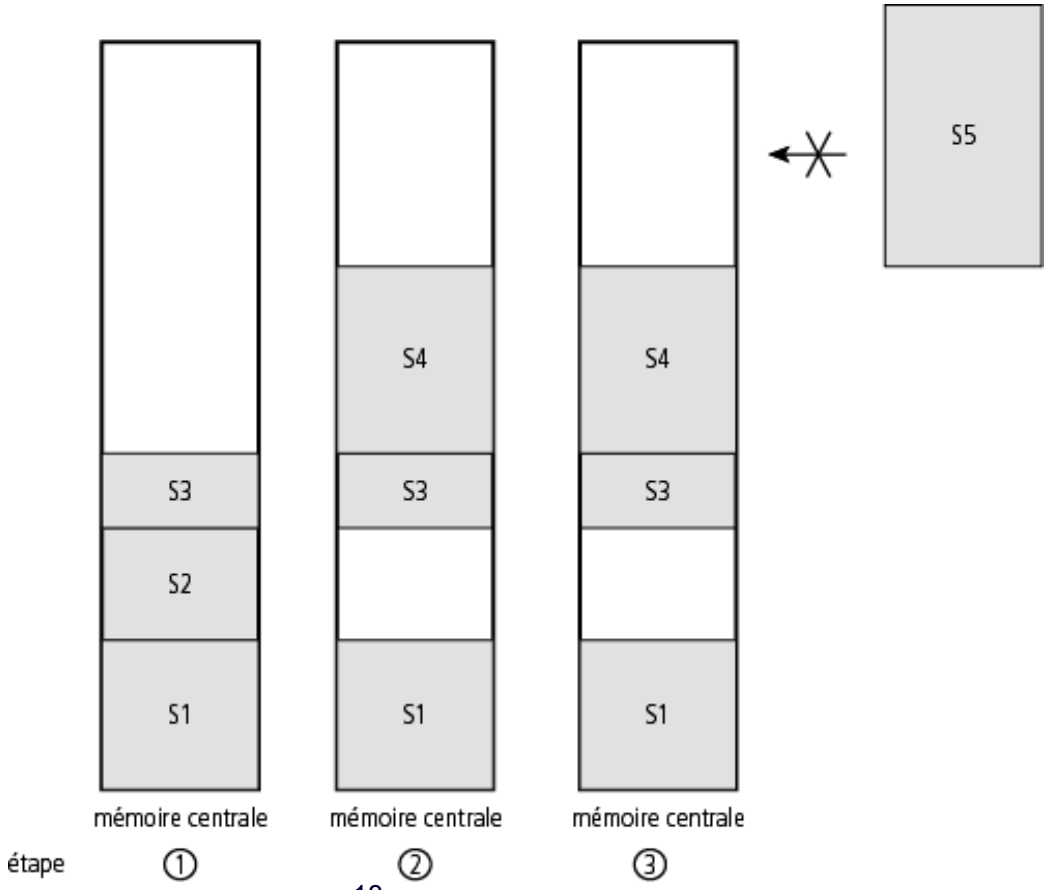


Quatre avantages

- Découpage d'un programme en segments
  - présence non obligatoire de tous les segments en MC
    - augmentation du degré de multiprogrammation
- La translation
  - les adresses logiques sont indépendantes de la position du segment en mémoire physique
    - augmentation du degré de multiprogrammation
- La protection
- Le partage par spécialisation du contenu
  - un segment peut être partagé (factorisation du code)
    - code réentrant

Défaut principal

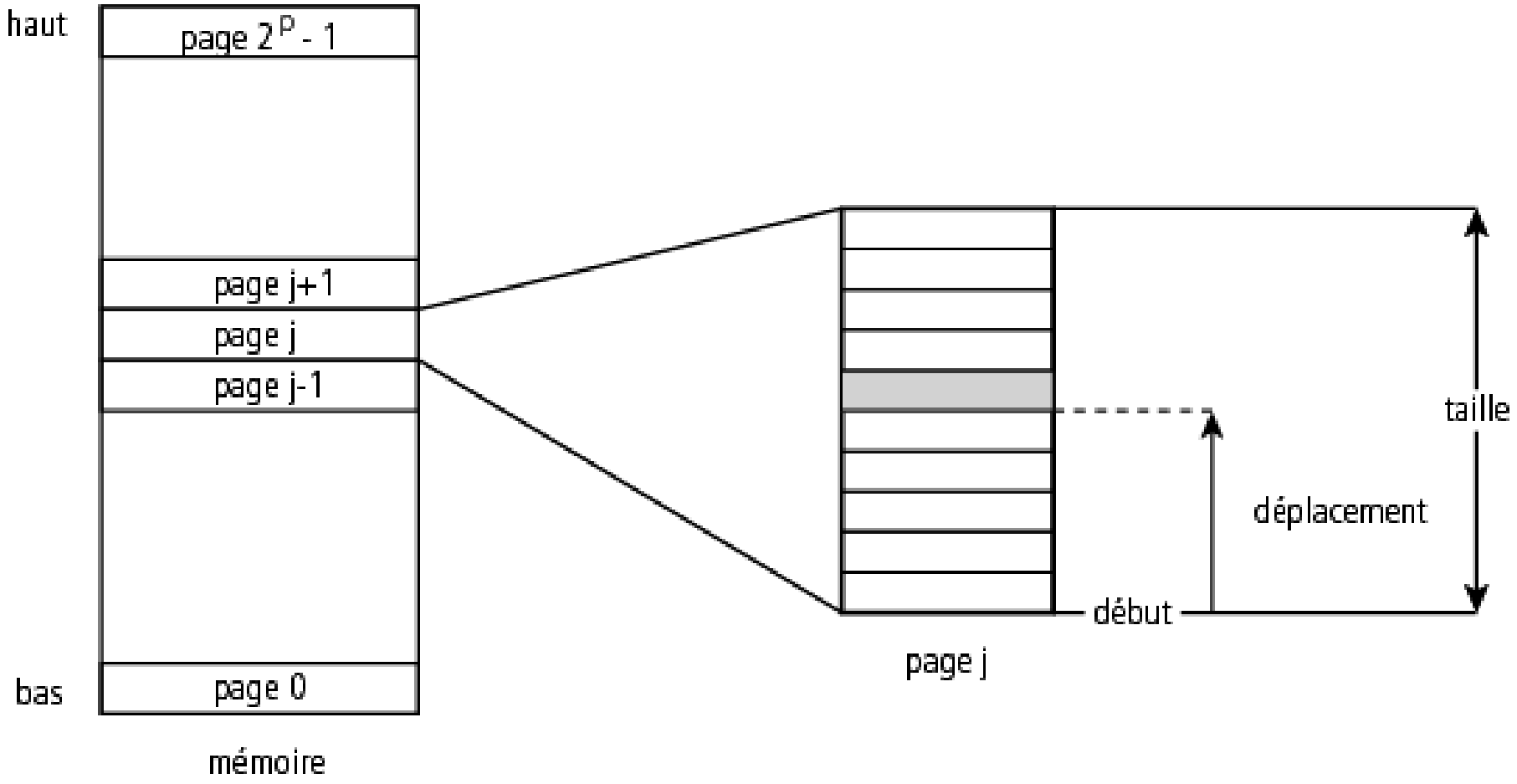
- La fragmentation de la mémoire centrale
- Un scénario



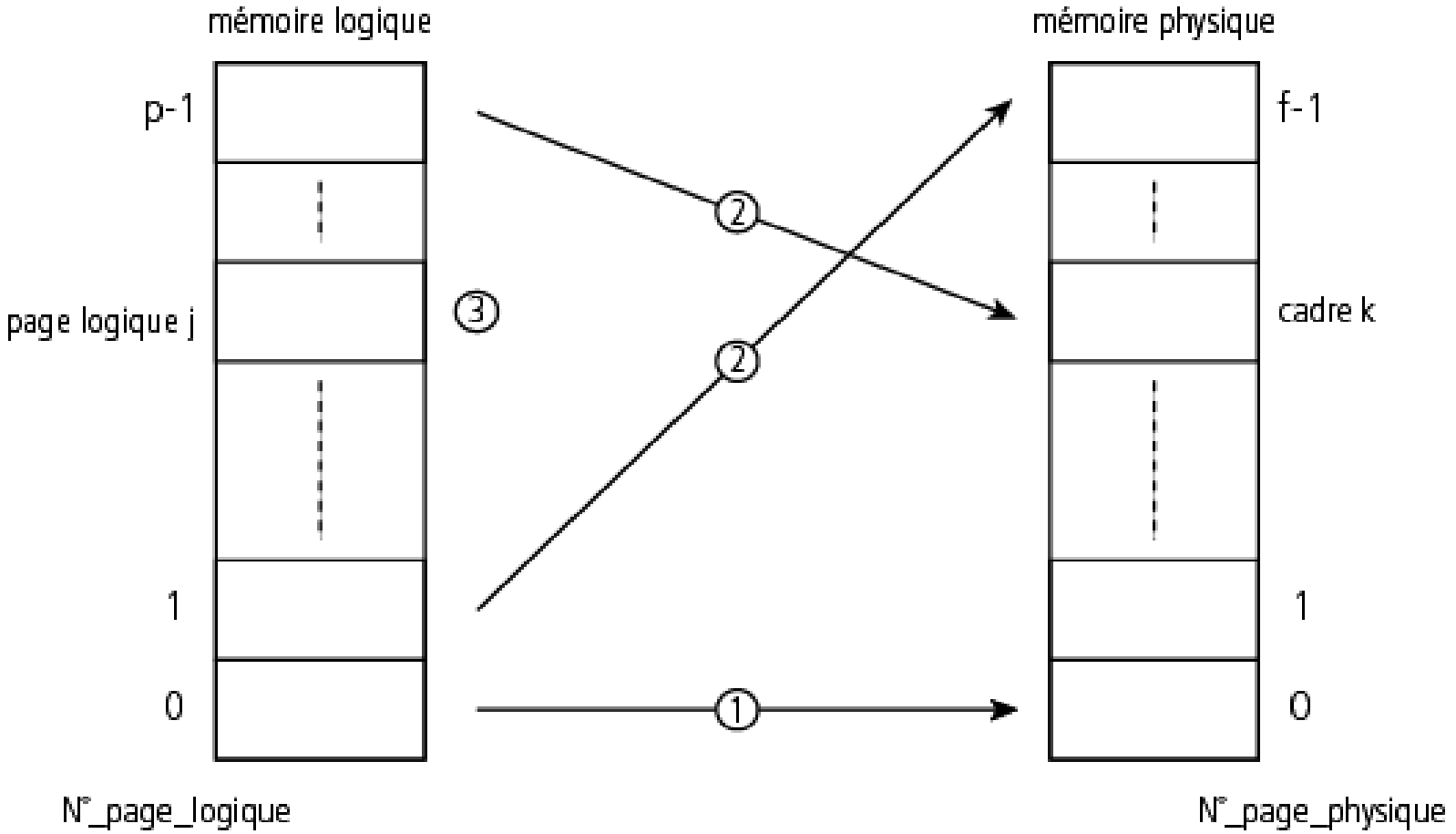
La pagination

- Une page
  - zone contigüe de mémoire
  - taille fixée par le matériel et le SE
    - ordre de grandeur : 1, 4 ou 8 Kio
  - repéré en mémoire par un indice n°_page
  - chaque mot est adressable par rapport au début de la page
    - adressage relatif
  - accueillie dans un cadre (*frame*) en mémoire centrale

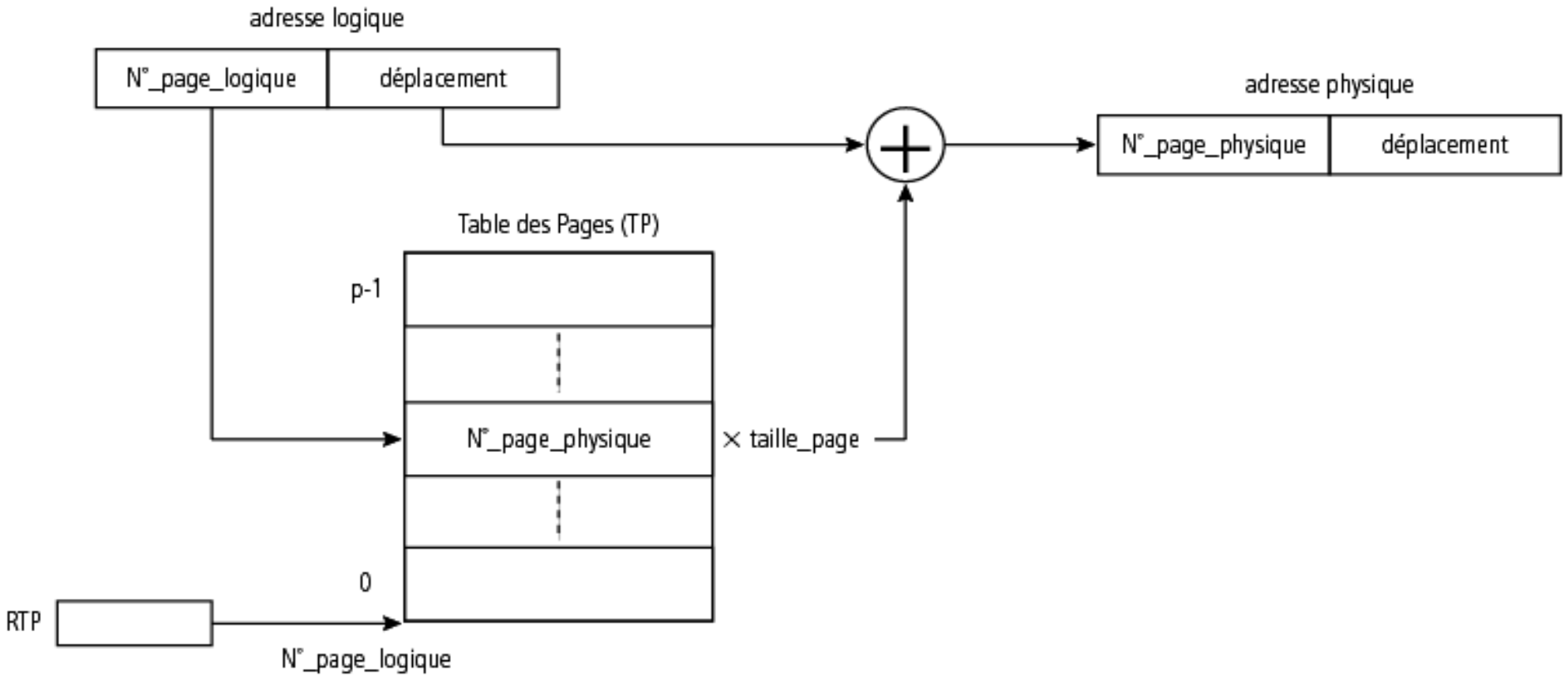
Disposition



Situations possibles



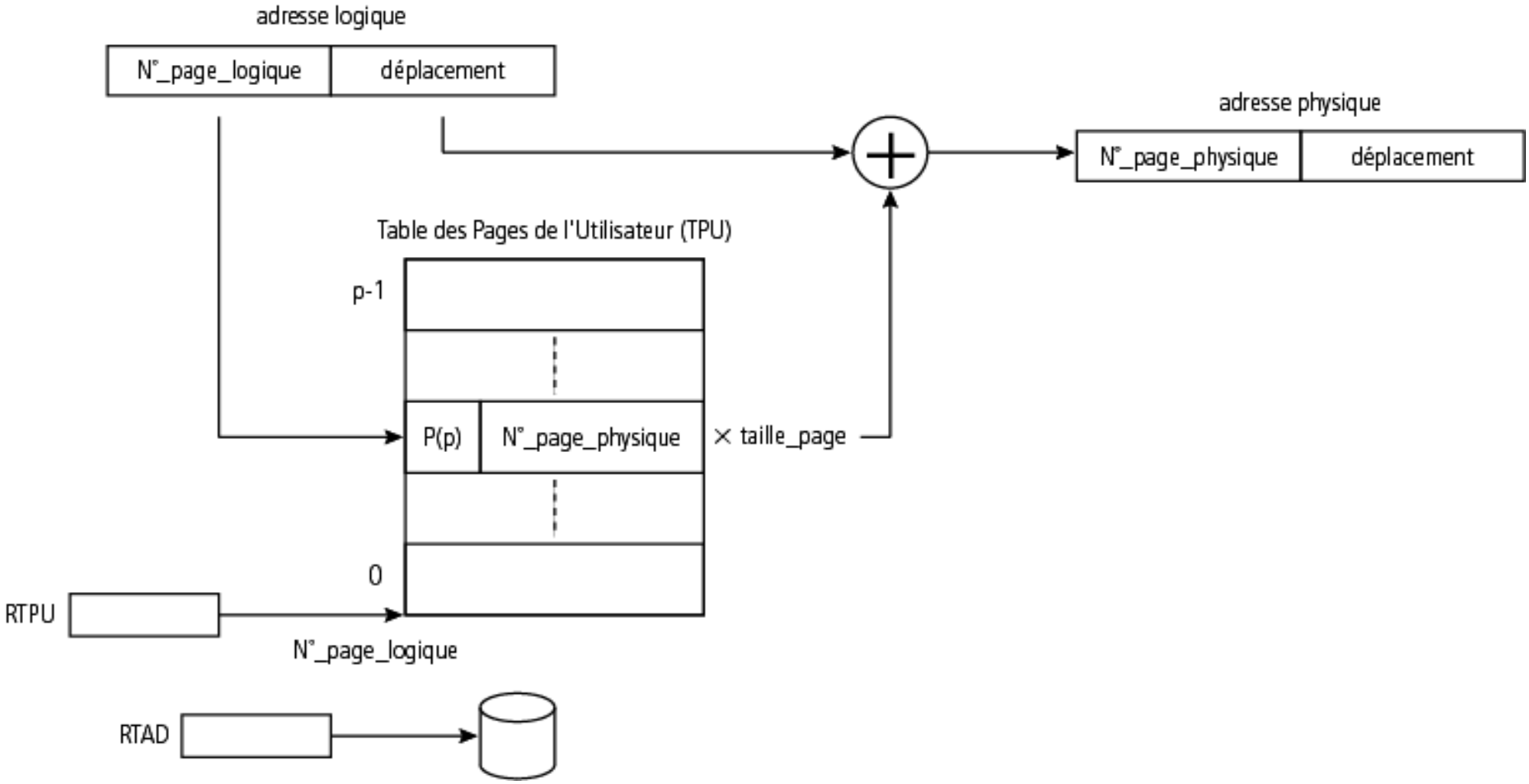
Calcul de l'adresse physique



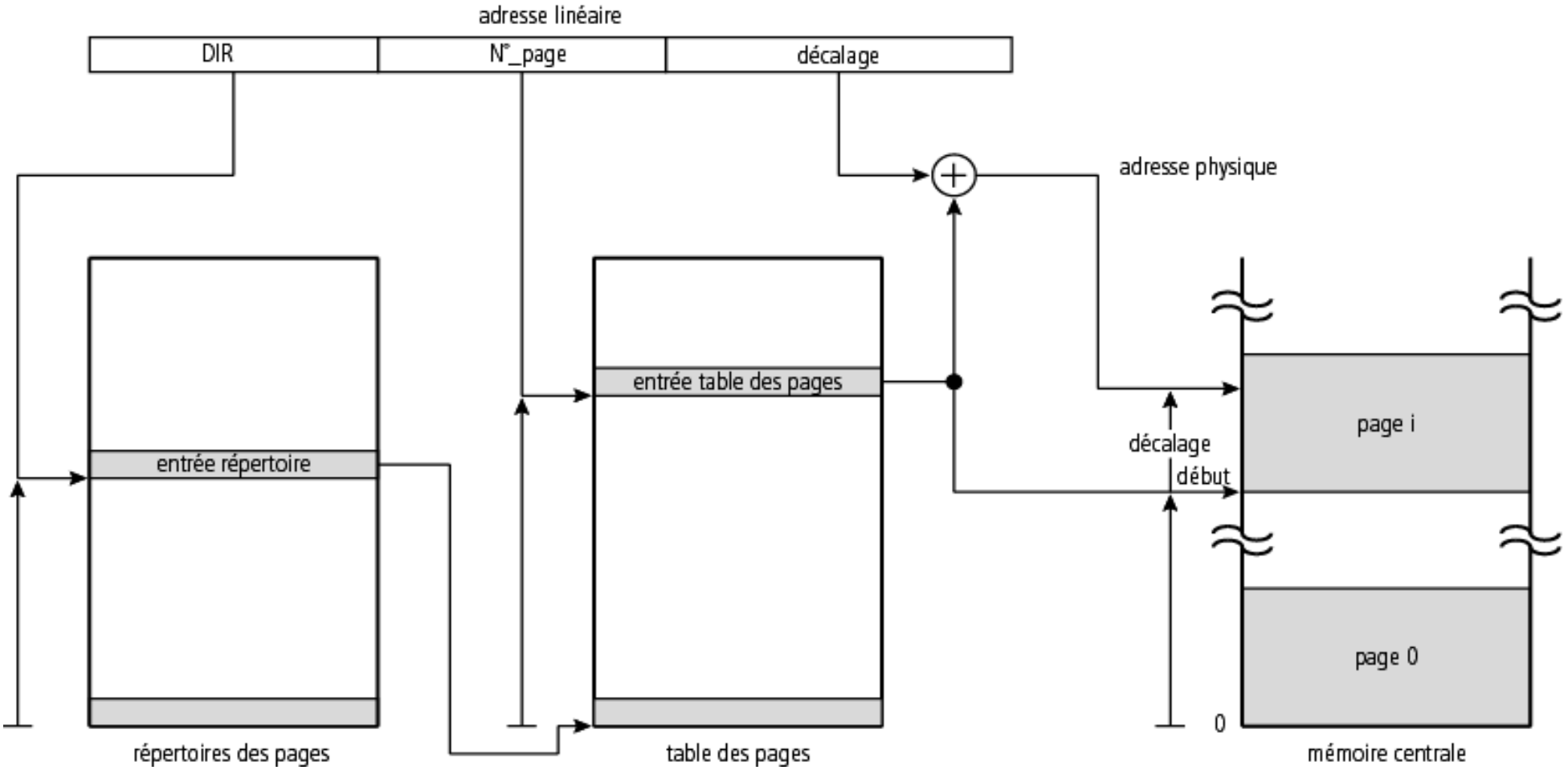
Autres fonctions du MMU

- Si la page physique n'est pas chargée,
  - faute de page (*page fault*)
  - génération d'une interruption par le MMU
- Vérification des droits d'accès
  - lecture/écriture/exécution

Mécanisme de protection



Gestion des pages du i386



Trois avantages

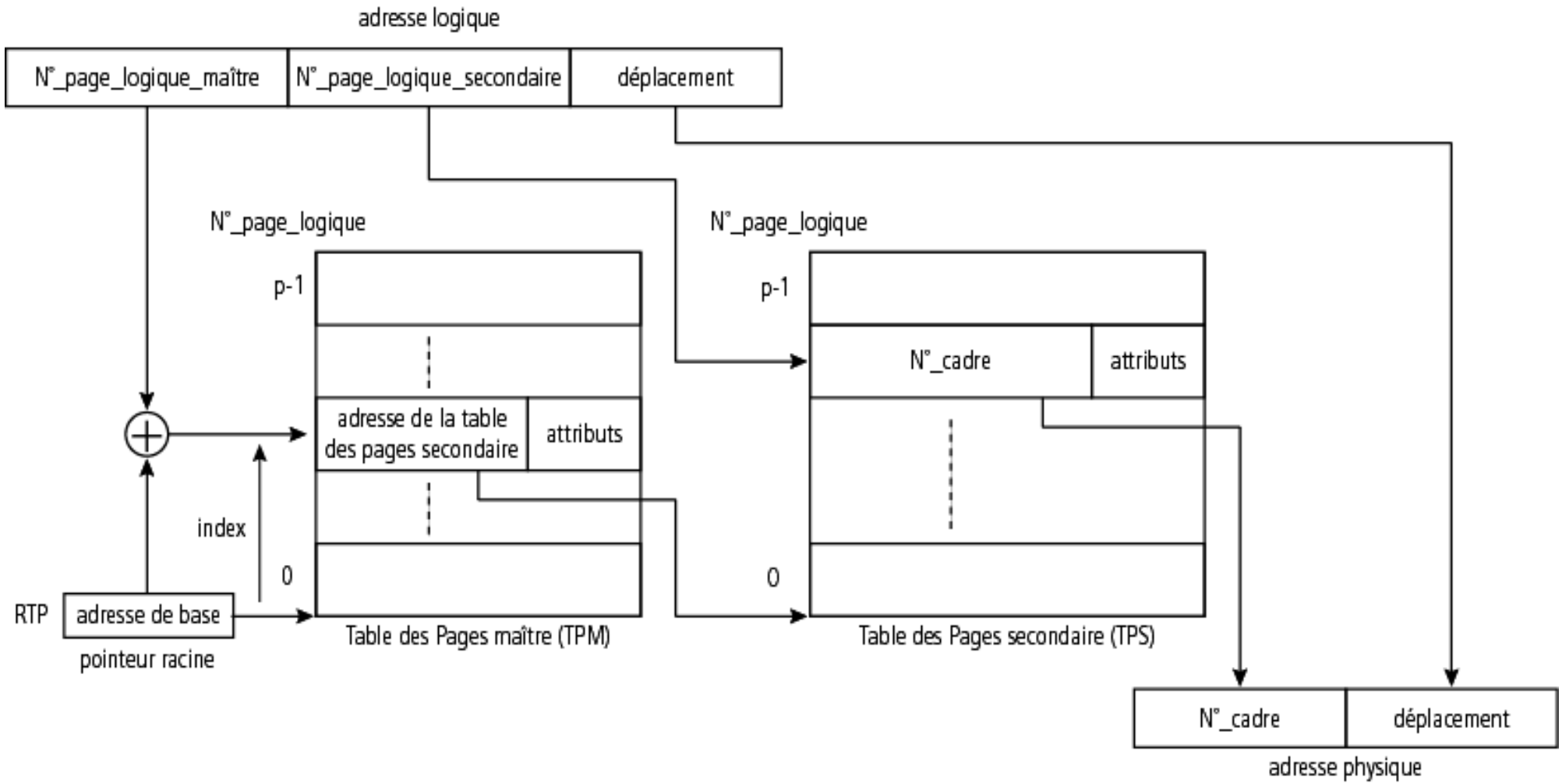
- Découpage d'un programme en pages
  - présence non obligatoire de toutes les pages en MC
    - augmentation du degré de multiprogrammation
- Meilleure gestion de la MC
  - problème de la fragmentation externe de la MC résolu
  - gestion aisée car taille fixe
- La protection

Problèmes à résoudre

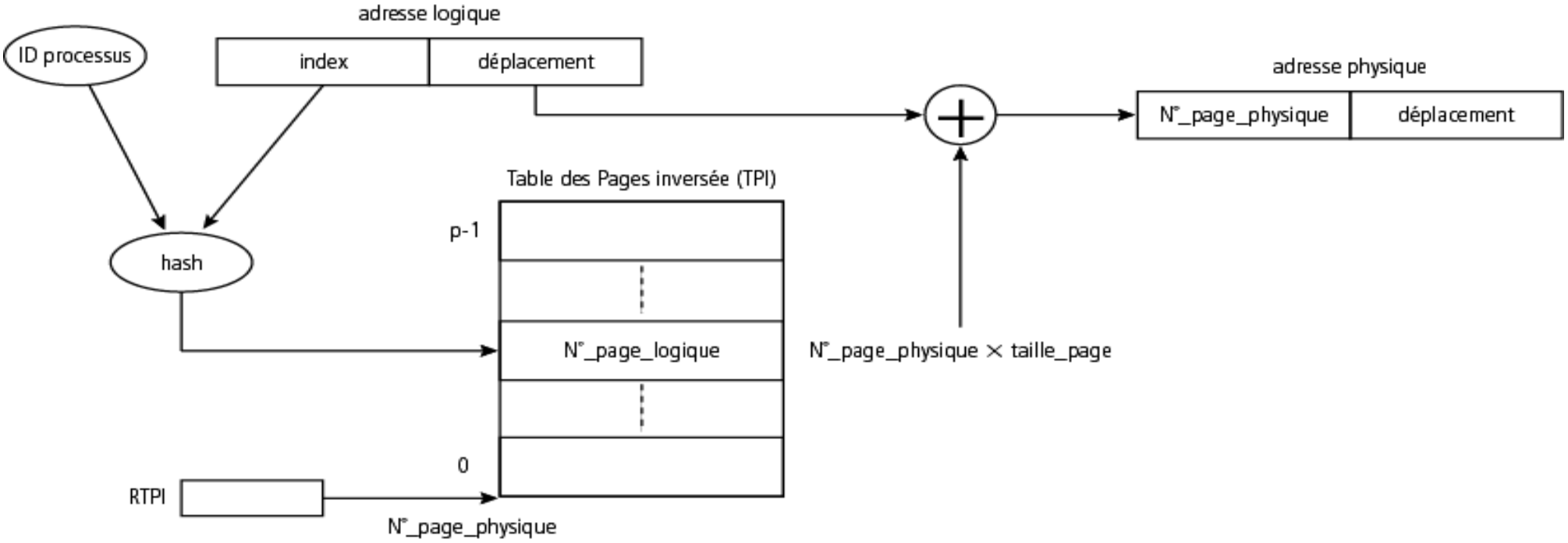
- Problème de la pagination
  - le nombre de pages
    - un espace d'adressage de 32 bits (4 Gio) peut représenter 1 Mi pages de 4 Kio

  Ce qui implique une très grande table des pages
- Solutions
  - tables des pages hiérarchisées
  - utilisation d'un cache
  - la pagination inversée

Les tables des pages hiérarchisées



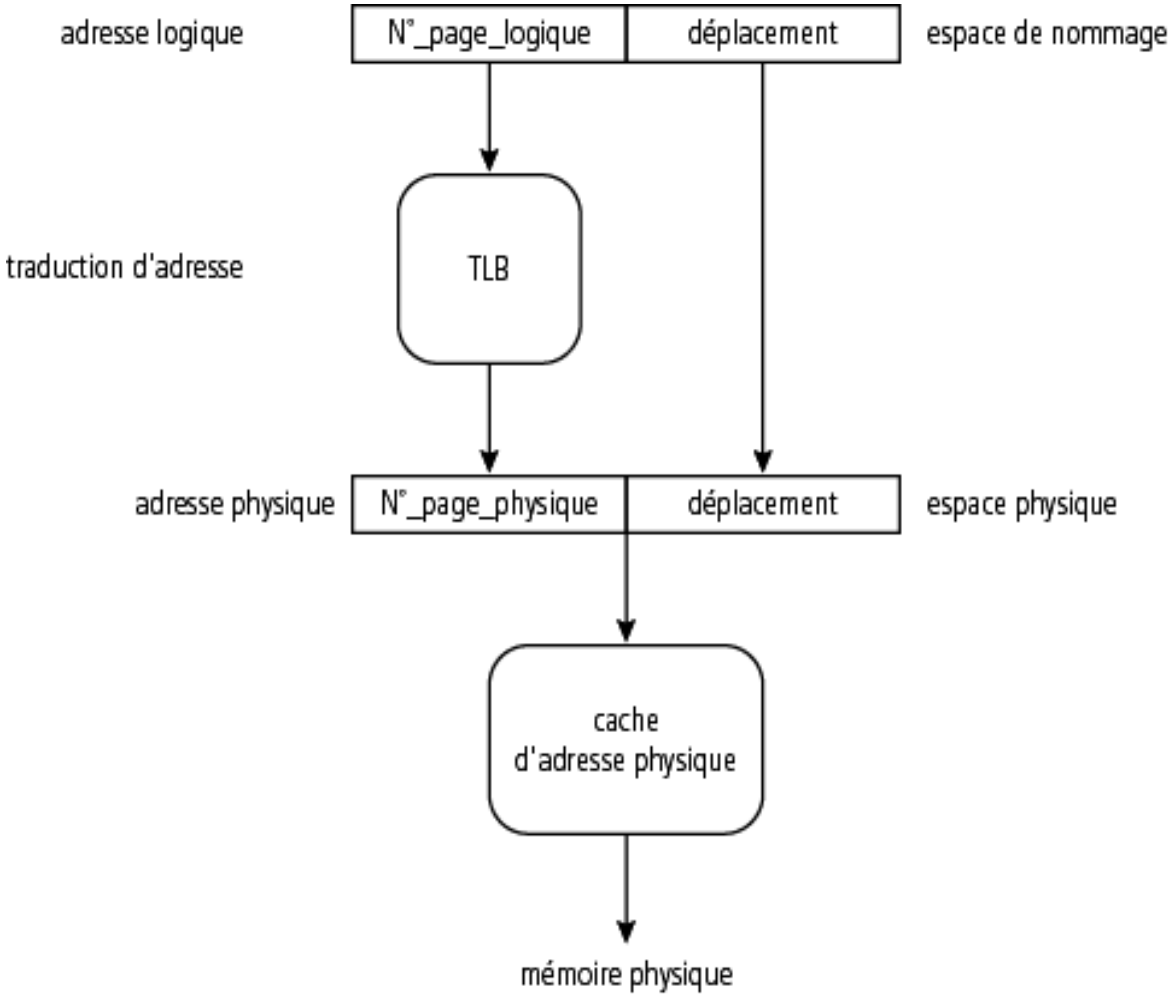
La pagination inversée



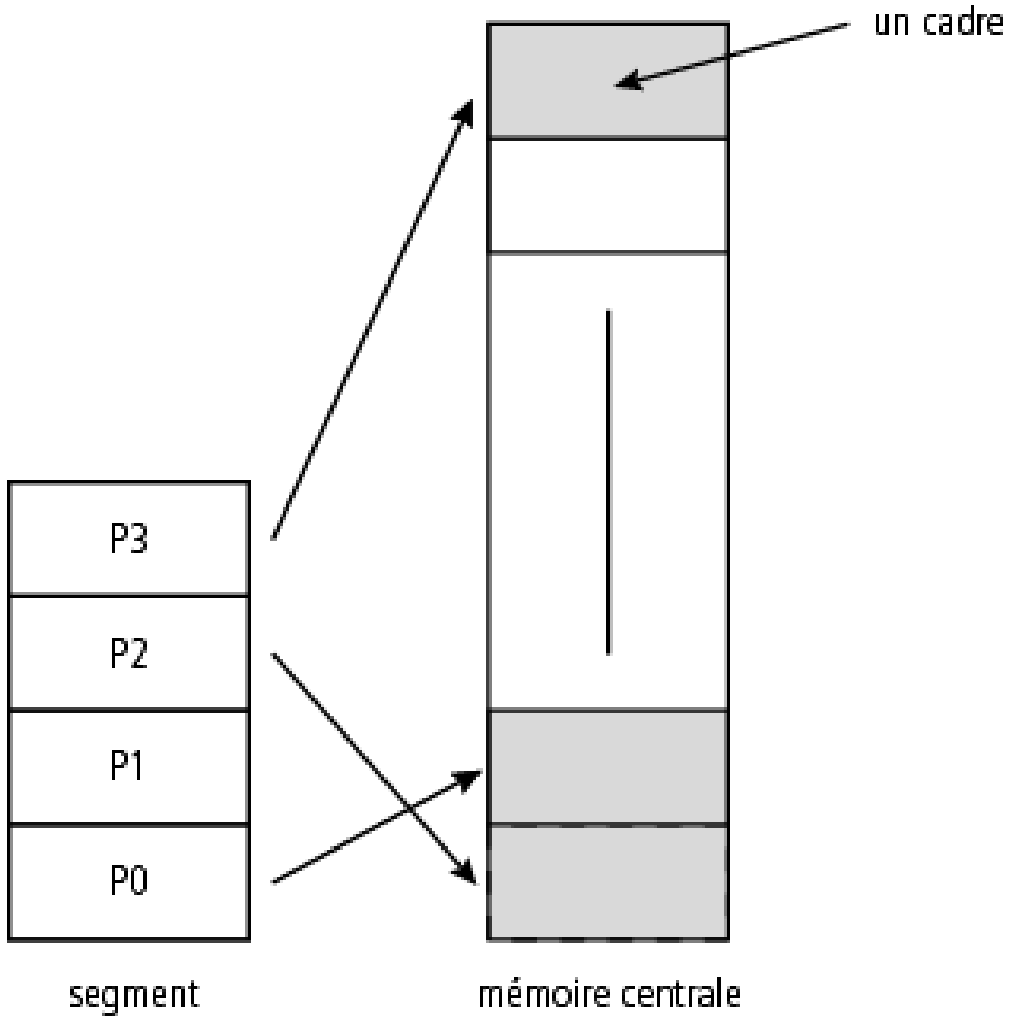
La TLB

- *Translation Look Aside Buffer*
- Utilisation d'un cache pour les adresses virtuelles
  - évite la traduction d'adresse
- Les processeurs modernes 64 bits en utilisent une

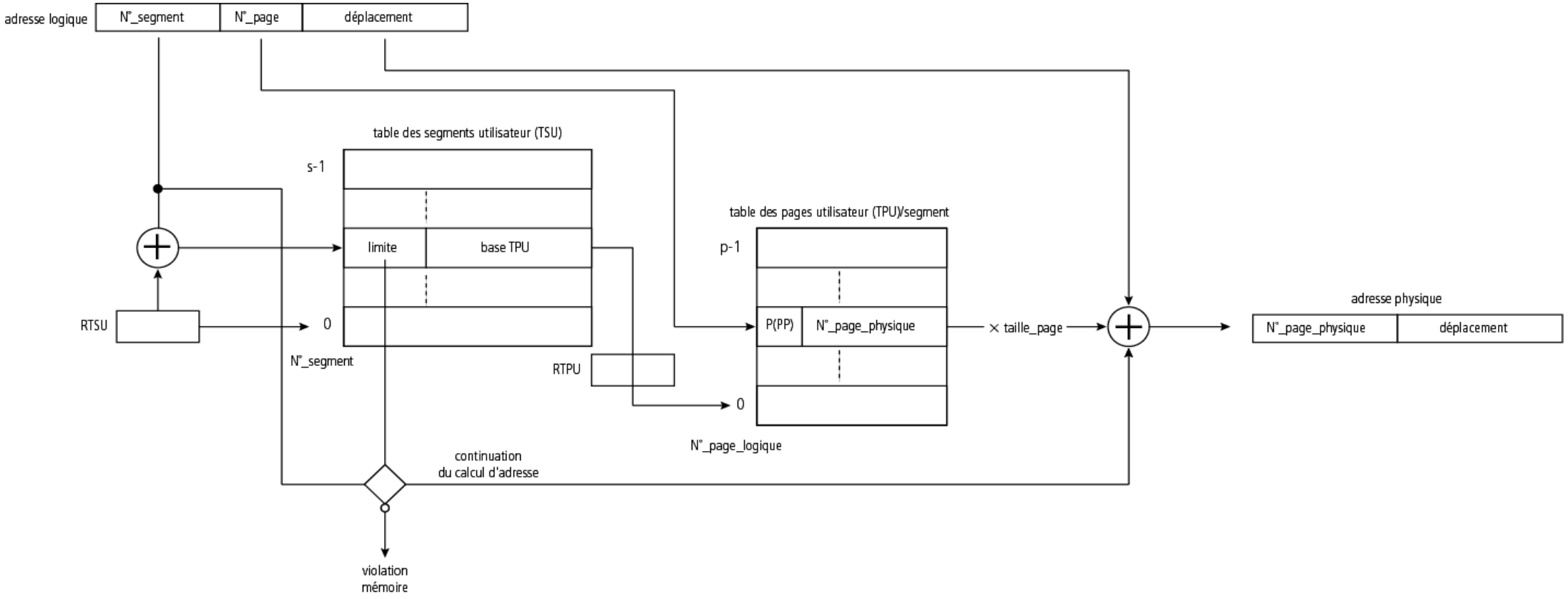
TLB pipelinée



Segmentation/pagination



Traduction d'adresse



Rôles du SE

- Chargement/remplacement/suppression de zones mémoires
- Le calcul de l'adresse physique est en général réalisé par le matériel sauf en cas d'échec d'accès à la TLB

Conclusion

- Mécanisme primordial dans les ordinateurs actuelles
- Offre :
  - un espace logique plus grand que l'espace physique
  - la protection de la mémoire
  - la translation transparente de zones de mémoires