

ΘΕΜΑ 3

Εστω 3 διεργασίες A, B και Γ των 16 σελίδων σε ένα σύστημα σελιδοποιημένης μνήμης (paging) το οποίο διαθέτει 32 πλαίσια σελίδων για χώρο εκτέλεσης διεργασιών (δηλαδή για κώδικα/δεδο-μένα/βιβλιοθήκες κ.λ.π. διεργασιών) με δυνατότητα **διαμοιρασμού** σελίδων/πλαισίων, οι οποίες κατά την εκτέλεσή τους,

- έχουν τους ακόλουθους πίνακες σελίδων η κάθε μία

- διαμοιράζονται μέσω αυτών τον κώδικα μίας βιβλιοθήκης 3 συναρτήσεων

- (α) Σε ποιά πλαίσια της φυσικής μνήμης έχει αντιστοιχηθεί ο κώδικας των 3 διαμοιραζόμενων συναρτήσεων βιβλιοθήκης;
- (β) Πώς θα άλλαζε το συνολικό σχήμα διευθυνσιοδότησης αν το σύστημα υποστήριζε σελιδοποιημένη τμηματοποίηση (paged segmentation), ομαδοποιώντας τις συναρτήσεις βιβλιοθήκης σε ένα τμήμα; Δώστε τους απαιτούμενους πίνακες τμημάτων ανά διεργασία και πίνακες σελίδων ανά τμήμα. Δώστε επίσης παράδειγμα αναφοράς διεύθυνσης για κάθε διεργασία σε θέση/byte του πλαισίου 25 της φυσικής μνήμης, με απλή σελιδοποίηση και με σελιδοποιημένη τμηματοποίηση αντίστοιχα.
- (γ) Τί όφελος θα είχαμε στην περίπτωση της σελιδοποιημένης τμηματοποίησης;
- (δ) Πώς θα διαμορφωνόταν το όφελος αυτό αν είχαμε N αντί για 3 διεργασίες τέτοιου είδους, καθώς και αντίστοιχα επαρκή διαθέσιμο χώρο φυσικής μνήμης;
- (ε) Τί μέγεθος/χώρος φυσικής μνήμης θα απαιτείτο σε μία τέτοια περίπτωση (N διεργασίες);

Πίνακας Σελίδων Α

0	3
1	1
2	4
3	31
4	28
5	2
6	29
7	12
8	15
9	25
10	18
11	19
12	8
13	10
14	11
15	22

Πίνακας Σελίδων Β

0	5
1	12
2	15
3	25
4	20
5	1
6	4
7	21
8	0
9	8
10	10
11	11
12	14
13	24
14	26
15	6

Πίνακας Σελίδων Γ

0	23
1	13
2	17
3	8
4	10
5	11
6	9
7	30
8	16
9	1
10	4
11	7
12	12
13	15
14	25
15	27

- (α) ο κώδικας των 3 διαμοιραζόμενων συναρτήσεων βιβλιοθήκης έχει αντιστοιχηθεί στα πλαίσια 1,4 (πρώτη συνάρτηση), 12, 15, 20 (δεύτερη συνάρτηση) και 8, 10, 11 (τρίτη συνάρτηση), τα οποία φαίνονται σκιασμένα παρακάτω (κοινή αντιστοίχιση και για τις τρεις διεργασίες):

Πίνακας Σελίδων Διεργ. Α		Πίνακας Σελίδων Διεργ. Β		Πίνακας Σελίδων Διεργ. Γ	
0	3	0	5	0	23
1	1	1	12	1	13
2	4	2	15	2	17
3	31	3	25	3	8
4	28	4	20	4	10
5	2	5	1	5	11
6	29	6	4	6	9
7	12	7	21	7	30
8	15	8	0	8	16
9	25	9	8	9	1
10	18	10	10	10	4
11	19	11	11	11	7
12	8	12	14	12	12
13	10	13	24	13	15
14	11	14	26	14	25
15	22	15	6	15	27

(β) αν είχαμε σελιδοποιημένη τμηματοποίηση, θα ήταν εφικτό:

- οι 3 συναρτήσεις βιβλιοθήκης να οργανωθούν σε 1 διαμοιραζόμενο τμήμα (8 συνολικά σελίδων/πλαισίων), κοινά προσπελάσιμο και από τις 3 διεργασίες,
- ο υπόλοιπος χώρος κώδικα/δεδομένων που απαιτεί κάθε μία από τις 3 διεργασίες να οργανωθεί και αυτός σε 1 τμήμα (ξεχωριστό – των 8 πλαισίων επίσης) ανά διεργασία.

Ως αποτέλεσμα κάθε διεργασία θα αποτελείτο από 2 τμήματα (0 και 1) και κατά συνέπεια θα απαιτούσε έναν Πίνακα Τμημάτων 2 θέσεων/εγγραφών για την περιγραφή αυτών

- Το πρώτο τμήμα της κάθε διεργασίας θα αντιστοιχούσε στον κώδικα και δεδομένα της διεργασίας, θα είχε μέγεθος 8 σελίδες και θα απαιτούσε έναν ξεχωριστό Πίνακα Σελίδων 8 θέσεων/εγγραφών
- Το δεύτερο τμήμα της κάθε διεργασίας θα αντιστοιχούσε στο 'κοινό' και για τις 3 διεργασίες τμήμα βιβλιοθήκης και θα απαιτούσε έναν ακόμη Πίνακα Σελίδων 8 θέσεων/εγγραφών (τον οποίον θα χρησιμοποιούν από κοινού και οι 3 διεργασίες).

Το συνολικό νέο σχήμα διευθυνσιοδότησης θα έχει ως ακολούθως:

Πίν. Τμημάτων Διεργ. Α Πίν. Τμημάτων Διεργ. Β Πίν. Τμημάτων Διεργ. Γ

	Οριο	Βάση		Οριο	Βάση		Οριο	Βάση
0	7	Διεύθ.Α	0	7	Διεύθ.Β	0	7	Διεύθ.Γ
1	7	Διεύθ.Δ	1	7	Διεύθ.Δ	1	7	Διεύθ.Δ

Πίν. Σελ. Τμήματος 0 Διεργ. Α

Πίν. Σελ. Τμήματος 0 Διεργ. Β

Πίν. Σελ. Τμήματος 0 Διεργ. Γ

(αρχίζει στη Διεύθ.Α)

(αρχίζει στη Διεύθ.Β)

(αρχίζει στη Διεύθ.Γ)

0	3	0	5	0	23
1	31	1	20	1	13
2	28	2	21	2	17
3	2	3	0	3	9
4	29	4	14	4	30
5	18	5	24	5	16
6	19	6	26	6	7

7	22	7	6	7	27
---	----	---	---	---	----

**Πίνακας Σελίδων Κοινού Τμήματος
(τμήμα 1) των Τριών Διεργασιών
(αρχίζει στη Διεύθ.Δ)**

0	1
1	4
2	12
3	15
4	25
5	8
6	10
7	11

Παράδειγμα αναφοράς μνήμης στο πλαίσιο 25

α. με απλή σελιδοποίηση

Κάθε λογική διεύθυνση (οποιασδήποτε από τις 3 διεργασίες) θα αποτελείτο από Αριθμό Λογικής Σελίδας 4-bits (καθώς ο λογικός χώρος διευθύνσεων της αποτελείται από $16=2^4$ σελίδες) και Μετατόπιση d-bits (όπου υποθέτουμε $P=2^d$ =μέγεθος σελίδας – το οποίο δεν μας δίνεται – υποθέτουμε χάριν του παραδείγματος $d=3/P=2^3=8$ bytes ανά σελίδα)

Διεργασία A: 1001 xxx (όπου (α) 1001=9 είναι ο Αριθμός Λογικής Σελίδας και μας οδηγεί στη θέση/εγγραφή 9 του πίνακα σελίδων της διεργασίας A η οποία αντιστοιχίζεται μέσω αυτού στο πλαίσιο 25 και (β) 'xxx' είναι η θέση της σελίδας/πλαισίου 25 την οποία θέλουμε να προσπελάσουμε – π.χ. αν θέλουμε να προσπελάσουμε την τρίτη κατά σειρά θέση του πλαισίου τότε xxx=010)

Διεργασία B: 0011 xxx

Διεργασία Γ: 1110 xxx

β. με σελιδοποιημένη τμηματοποίηση

Κάθε λογική διεύθυνση (οποιασδήποτε από τις 3 διεργασίες) θα αποτελείτο από Αριθμό Τμήματος 1-bit, Αριθμό Σελίδας Τμήματος 3-bits και Μετατόπιση d-bits (ακριβώς όπως και στην περίπτωση της απλής σελιδοποίησης - υποθέτουμε $d=3$)

Διεργασία A: 1 100 xxx (όπου (α) 1 είναι ο Αριθμός Τμήματος που μας οδηγεί στη θέση/εγγραφή 1 του πίνακα τμημάτων της διεργασίας A από την οποία λαμβάνουμε τη διεύθυνση Βάσης – Διευθ.Δ – από όπου αρχίζει ο πίνακας σελίδων του συγκεκριμένου τμήματος, (β) 100 είναι η θέση/εγγραφή του πίνακα σελίδων του συγκεκριμένου τμήματος όπου πρέπει να πάμε και να λάβουμε το αντίστοιχο πλαίσιο αντιστοίχισης στη μνήμη – πλαίσιο 25, και (γ) 'xxx' είναι η θέση της σελίδας/πλαισίου 25 την οποία θέλουμε να προσπελάσουμε – π.χ. αν θέλουμε να προσπελάσουμε την τρίτη κατά σειρά θέση του πλαισίου τότε xxx=010)

Διεργασία B: 1 100 xxx

Διεργασία Γ: 1 100 xxx

Παρατηρούμε ότι και για τις 3 διεργασίες (με σελιδοποιημένη τμηματοποίηση) η λογική διεύθυνση αναφοράς σε πλαίσια του κοινού τμήματος θα είναι ίδια (αρκεί

το κοινό τμήμα να έχει αριθμηθεί ως ίδιο – 1 στην περίπτωση μας – για όλες τις διεργασίες)

(γ) και στις δύο περιπτώσεις (απλή σελιδοποίηση και σελιδοποιημένη τμηματοποίηση) απαιτείται ο ίδιος χώρος στη μνήμη (32 πλαίσια) για την εκχώρηση/αντιστοίχιση κώδικα, διαμοιραζόμενων συναρτήσεων και δεδομένων που απαιτούν οι 3 διεργασίες.

Το κέρδος (ως προς τη χρήση σελιδοποιημένης τμηματοποίησης) αφορά στο χώρο που απαιτείται **για την αποθήκευση των απαιτούμενων βοηθητικών πινάκων (πίνακες σελίδων, πίνακες τμημάτων κ.λ.π.)**, και πιο συγκεκριμένα:

- με απλή σελιδοποίηση απαιτούνται 3 Πίνακες Σελίδων των 16 εγγραφών = 48 εγγραφές
- με σελιδοποιημένη τμηματοποίηση απαιτούνται 3 Πίνακες Τμημάτων των 2 εγγραφών και 4 Πίνακες Σελίδων Τμημάτων των 8 εγγραφών = συνολικά (και αν χωρίς βλάβη της γενικότητας υποθέσουμε ότι το μήκος εγγραφής πίνακα τμήματος και πίνακα σελίδων είναι περίπου το ίδιο) 38 εγγραφές

(δ) αν είχαμε N αντί για 3 τέτοιες διεργασίες, τότε θα απαιτούντο:

- στην περίπτωση της απλής σελιδοποίησης: N Πίνακες σελίδων των 16 εγγραφών = συνολικά $16N$ εγγραφές
- στην περίπτωση της σελιδοποιημένης τμηματοποίησης: N Πίνακες Τμημάτων των 2 εγγραφών και $N+1$ Πίνακες Σελίδων Τμημάτων των 8 εγγραφών = συνολικά $8(N+1)+2N= 10N+8$ εγγραφές

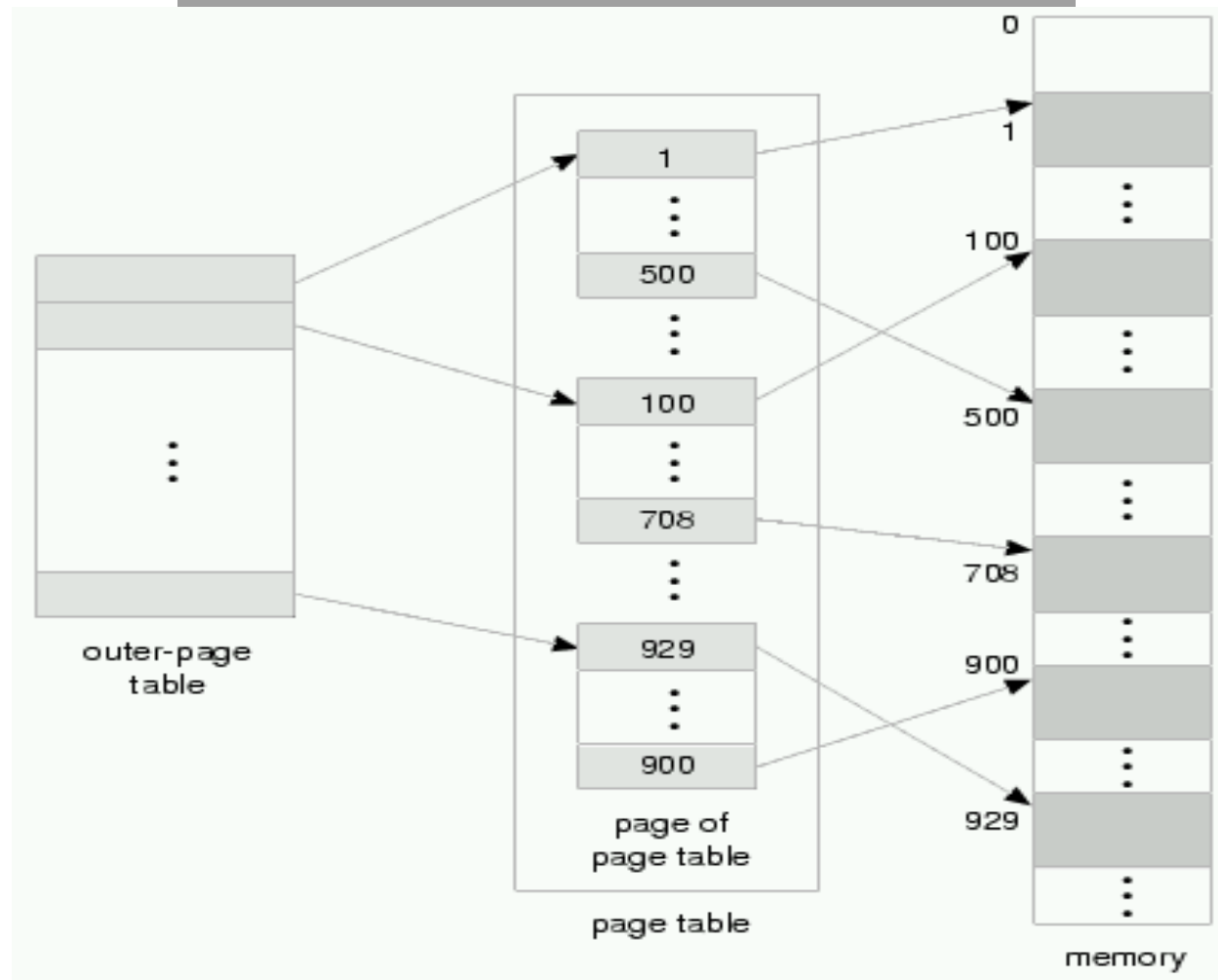
Ποιοτικά, το συνολικό κέρδος μπορεί να εκφραστεί ως ακολούθως:

Με σελιδοποιημένη τμηματοποίηση για κάθε επιπλέον ιδίων χαρακτηριστικών διεργασία που εισάγεται,

- κερδίζουμε το ήμισυ του Πίνακα Σελίδων που απαιτείται με απλή σελιδοποίηση (8 εγγραφές στη συγκεκριμένη περίπτωση ή γενικότερα τόσες εγγραφές όσες είναι οι σελίδες του κοινού τμήματος ή ακόμα γενικότερα τόσες εγγραφές όσες είναι οι σελίδες των κοινών τμημάτων αν είχαμε παραπάνω από ένα τέτοια) και
- επιβαρυνόμαστε με έναν επιπλέον Πίνακα Τμημάτων (2 εγγραφές στη συγκεκριμένη περίπτωση ή γενικότερα τόσες εγγραφές όσα είναι τα κοινά τμήματα συν ένα αν είχαμε παραπάνω από ένα τέτοια)
- όσο μεγαλύτερο(α) είναι το κοινό(ά) τμήμα(τα) των διεργασιών τόσο μεγαλύτερο κέρδος έχουμε σε χώρο αποθήκευσης.

(ε) Τόσο με απλή σελιδοποίηση όσο και με σελιδοποιημένη τμηματοποίηση ο χώρος στη μνήμη που θα απαιτείτο αν είχαμε N ιδίων χαρακτηριστικών διεργασίες, για την εκχώρηση/αντιστοίχιση του κώδικα, διαμοιραζόμενων συναρτήσεων και δεδομένων τους (εξαιρώντας δηλαδή το χώρο αποθήκευσης των βοηθητικών πινάκων σελίδων και τμημάτων ο οποίος υπολογίστηκε παραπάνω στο σημείο δ), θα ήταν ίσος με 8 σελίδες/πλαίσια ανά διεργασία συν 8 πλαίσια για το κοινό τμήμα, δηλαδή συνολικά $8(N+1)$ πλαίσια.

TwoLevel PageTable Scheme



TwoLevel Paging Example

- A logical address (on 32bit machine with 4K page size) is divided into:
 - a page number consisting of 20 bits.
 - a page offset consisting of 12 bits.
- Since the page table is paged, the page number is further divided into:
 - a 10bit page number.
 - a 10bit page offset.
- Thus, a logical address is as follows:

page number		page offset
p_1	p_2	d
10	10	12

where p_1 is an index into the outer page table, and p_2 is the displacement within the page of the outer page table.

Εστω ότι έχουμε ένα σύστημα μνήμης με σελιδοποίηση δύο επιπέδων (2-level paging):

- ο 32bit λογική διεύθυνση
- ο 10bit page number (p1)
- ο 10bit page offset (p2)
- ο 12bit final page offset (d)

Εστω η λογική διεύθυνση (σε δεκαεξαδικό σύστημα) → 00403004

Και ας υποθέσουμε ότι έχουμε τους ακόλουθους πίνακες σελίδων

Outer page table	
0	Διεύθυνση Α
1	Διεύθυνση Β
2	Διεύθυνση Γ
3	Διεύθυνση Δ
4	Διεύθυνση Ε
.....	
.....	

Διεύθυνση Α	
0	10001000
1	10101010
2	01010101
3	01010000
4	00000001
.....
.....

Διεύθυνση Β	
0	00010001
1	00100100
2	11011011
3	00001111
4	11110000
.....
.....

Σε ποιά φυσική διεύθυνση αντιστοιχεί η παραπάνω λογική διεύθυνση (00403004)?

Απάντηση:

Η δεκαεξαδική λογική διεύθυνση 00403004 αντιστοιχεί στο δυαδικό σε

0000-0000-0100-0000-0011-0000-0000-0100 ή αλλιώς (αν την χωρίσουμε σε 10-10-12 bits)

0000000001-0000000011-000000000100, το οποίο μας υπαγορεύει

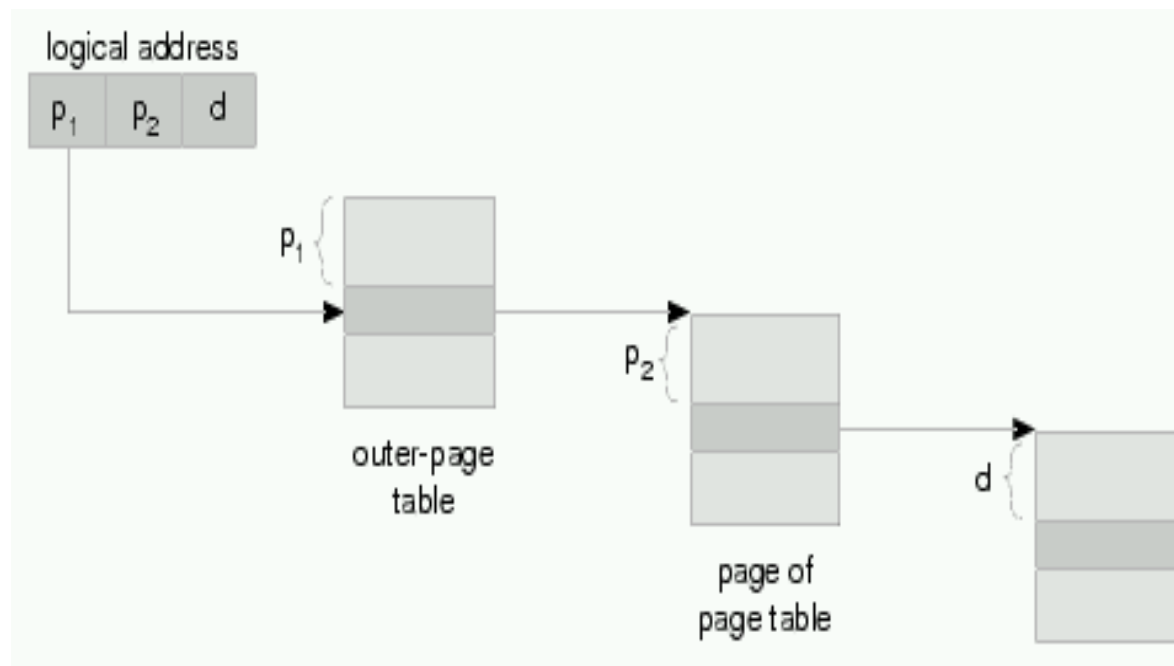
α. να πάμε στη θέση 0000000001 (1) του outer table και να πάρουμε από εκεί τη διεύθυνση αρχής του επιμέρους πίνακα σελίδων (Διεύθυνση Β)

β. να πάμε στη θέση 0000000011 (3) του επιμέρους αυτού πίνακα σελίδων και να πάρουμε από κει τον αριθμό φυσικού πλαισίου '00001111'

γ. να προσθέσουμε σε αυτόν το final-page-offset 000000000100 και έτσι να καταλήξουμε σσε τελική φυσική διεύθυνση 00001111-000000000100 ή αλλιώς 0F004 (δεκαεξαδικό)

Address Translation Scheme

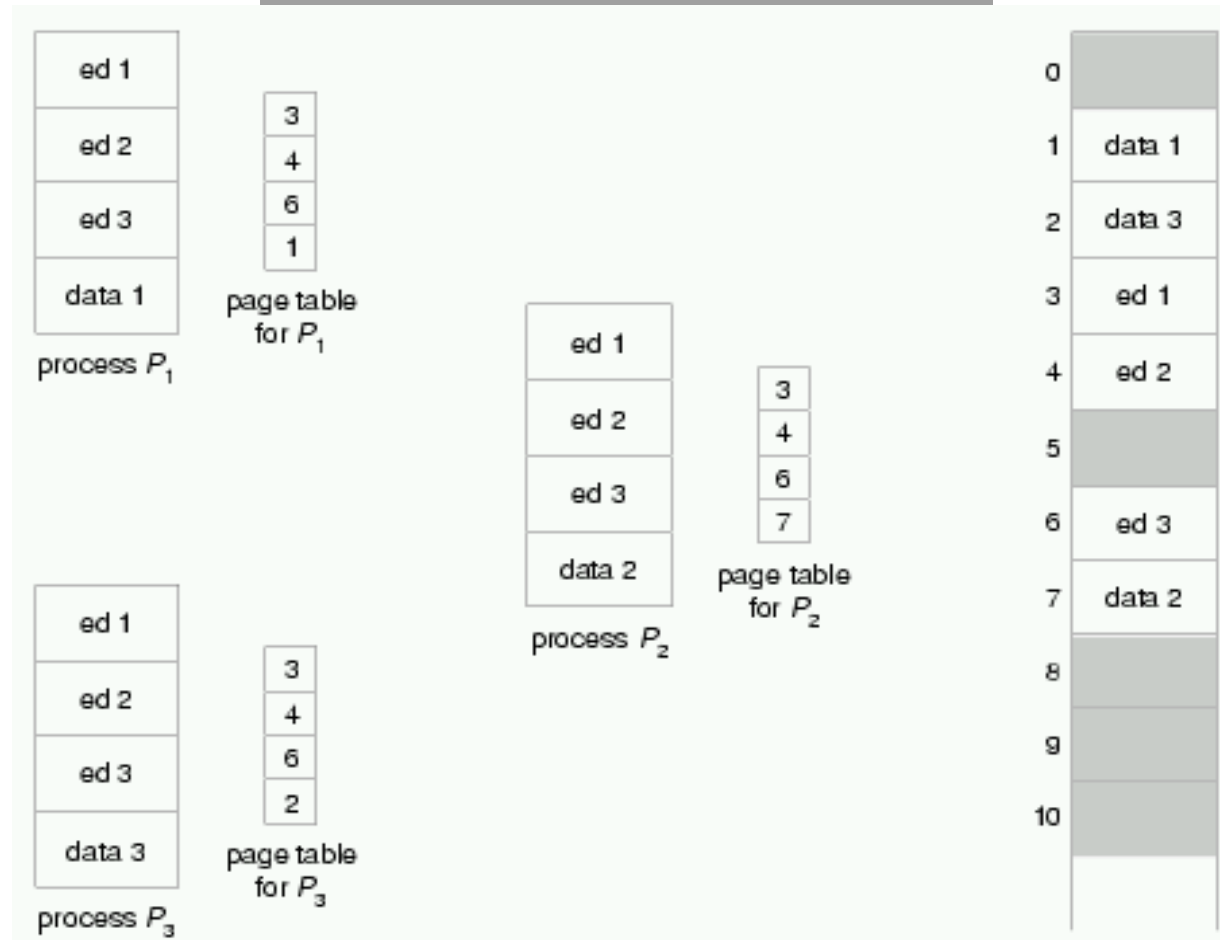
- Address translation scheme for a two level 32bit paging architecture



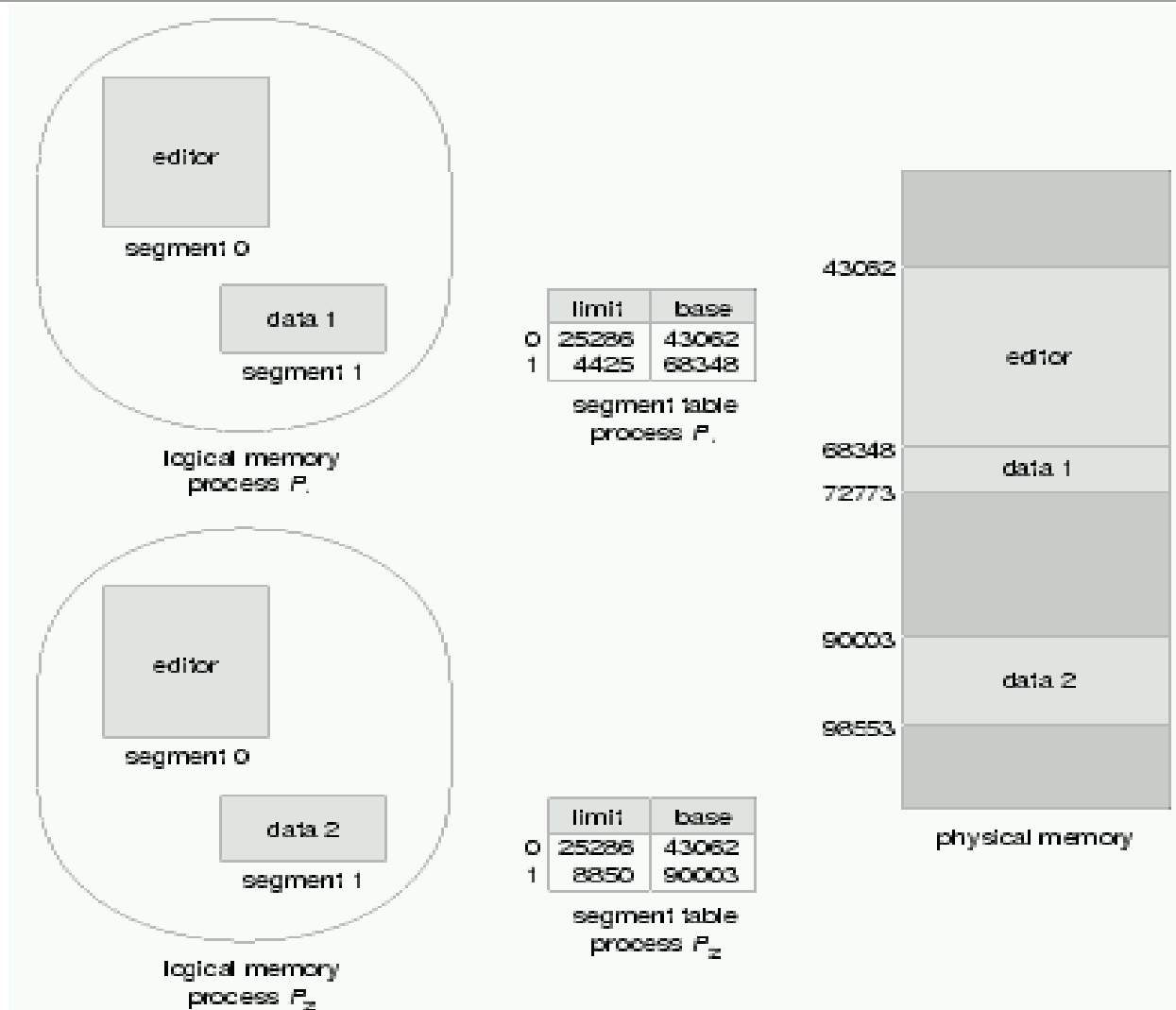
Shared Pages

- Shared code
 - One copy of readonly (reentrant) code shared among processes (i.e., text editors, compilers, window systems).
 - Shared code must appear in same location in the logical address space of all processes.
- Private code and data
 - Each process keeps a separate copy of the code and data
 - the pages for the private code and data can appear anywhere in the logical address space

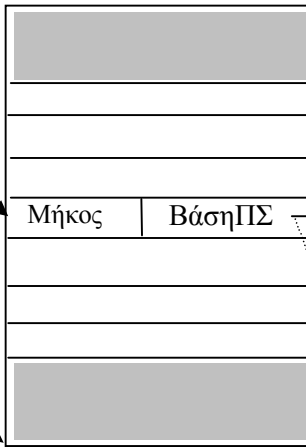
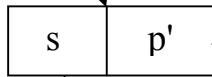
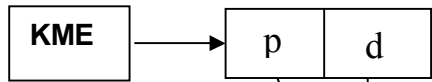
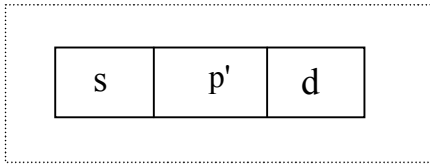
Shared Pages Example



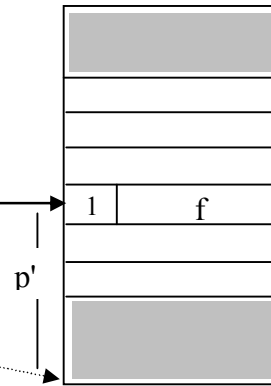
Shared Segments Example



SEGMENTED PAGING
(τμηματοποίηση του
Πίνακα Σελίδων)



Πίνακας Τμημάτων



Τμήμα του
Πίνακα Σελίδων

