

Οργάνωση Βάσεων Βιοϊατρικών Δεδομένων Εξόρυξη Γνώσης Βιοϊατρικών Δεδομένων

Σεμινάριο 6: Δομές ευρετηρίων για αρχεία

**Ευάγγελος Καρκαλέτσης,
Αναστασία Κριθαρά, Γεώργιος Πετάσης**

Εργαστήριο Τεχνολογίας Γνώσεων & Λογισμικού,

Ινστιτούτο Πληροφορικής & Τηλεπικοινωνιών, Ε.Κ.Ε.Φ.Ε. “Δημόκριτος”

Τηλ.: 210-6503197, Fax: 210-6532175, {vangelis, akrihara, petasis}@iit.demokritos.gr

Ακαδημαϊκό Έτος: 2016 – 2017

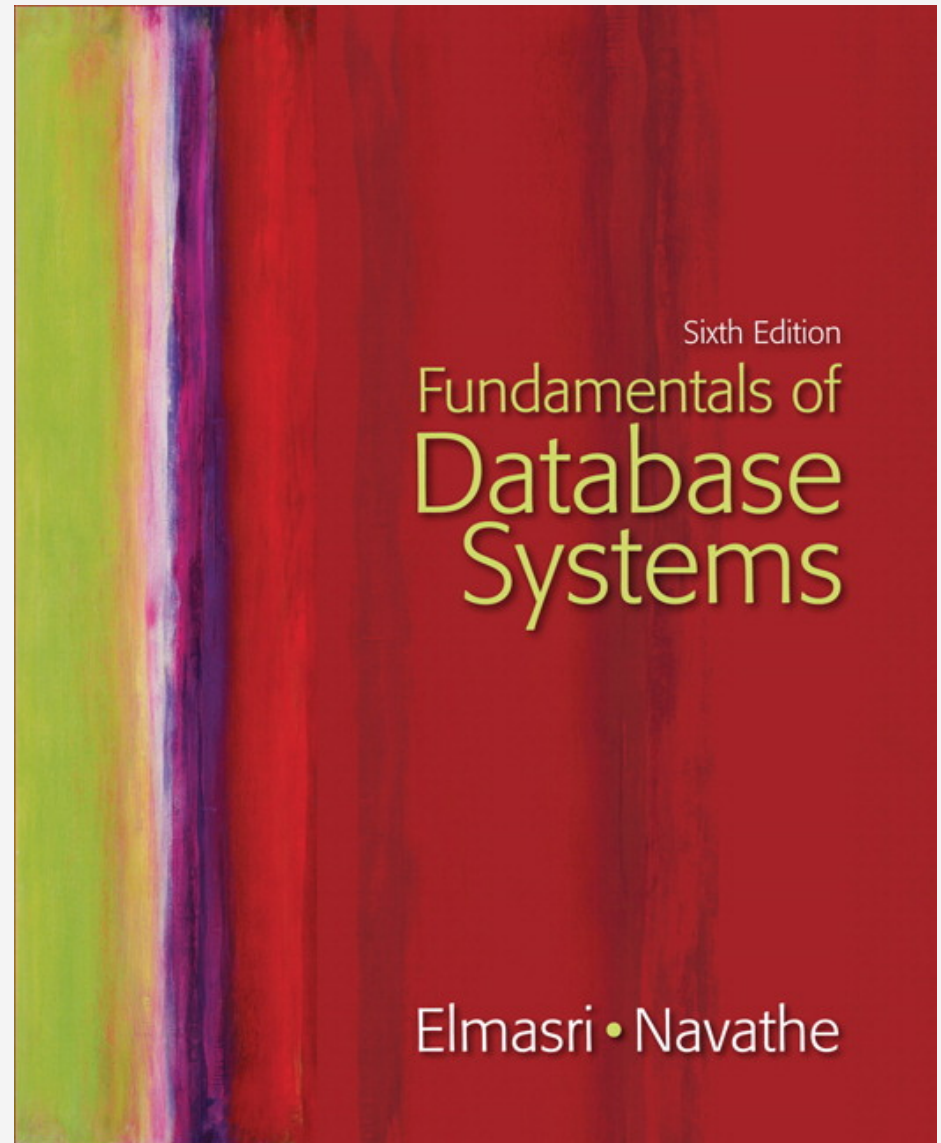
Οι διαφάνειες αυτού του σεμιναρίου βασίζονται
σε διαφάνειες των:

Δ. Γουνόπουλος, Ι. Ιωαννίδης

<http://eclass.uoa.gr/modules/document/document.php?course=D463&openDir=/4c2b32a5i4iq>

ΚΕΦΑΛΑΙΟ 18

Δομές ευρετηρίων για αρχεία



Copyright © 2011 Ramez Elmasri and Shamkant B. Navathe, Ελληνική Έκδοση Δίαυλος

Θα μιλήσουμε για

- Τύποι Ταξινομημένων Ευρετηρίων ενός επιπέδου
 - Πρωτεύοντα Ευρετήρια
 - Ευρετήρια Συστάδες
 - Δευτερεύοντα Ευρετήρια
- Ευρετήρια Πολλών επιπέδων
- Δυναμικά Ευρετήρια Πολλών Επιπέδων με χρήση B-Δένδρων και B⁺-Δένδρων
- Ευρετήρια σε Πολλαπλά Κλειδιά
- Ευρετήρια χωρικών δεδομένων με χρήση R-Δέντρων

Ευρετήρια σαν Μέθοδοι Προσπέλασης

- Ένα ευρετήριο ενός επιπέδου είναι ένα βοηθητικό αρχείο που κάνει πιο αποτελεσματική την αναζήτηση μιας εγγραφής σε ένα αρχείο δεδομένων.
- Το ευρετήριο συνήθως ορίζεται σε ένα πεδίο του αρχείου (αν και μπορεί να ορισθεί σε πολλά πεδία)
- Μια μορφή ευρετηρίου είναι ένα αρχείο με καταχωρήσεις **<τιμή πεδίου, δείκτης στην εγγραφή>**, που είναι ταξινομημένο με τιμή πεδίου
- Το ευρετήριο λέγεται δρόμος προσπέλασης στο πεδίο.

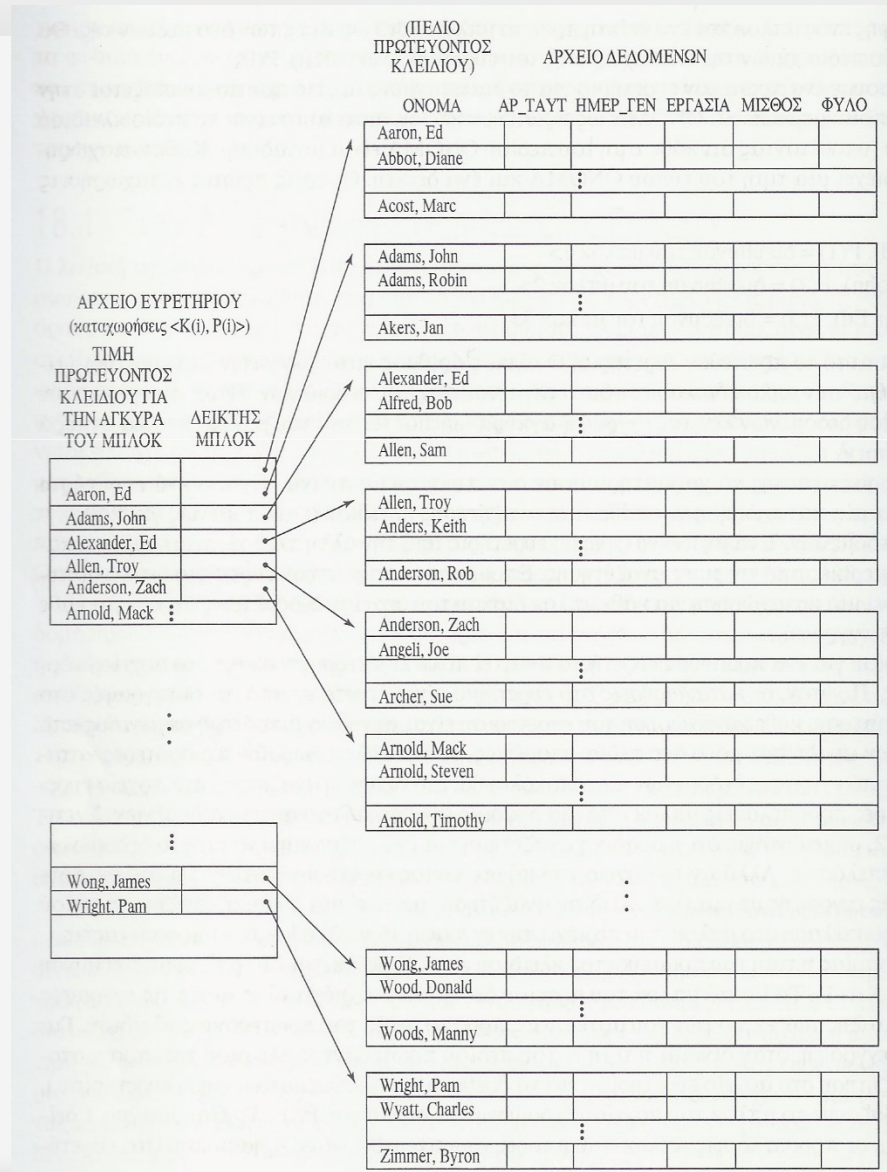
Ευρετήρια σαν Μέθοδοι Προσπέλασης (συνέχεια)

- Το αρχείο του ευρετηρίου συνήθως καταλαμβάνει σημαντικά μικρότερο πλήθος μπλοκ από ότι το αρχείο δεδομένων επειδή οι καταχωρήσεις του είναι κατά πολύ μικρότερες
- Μια δυαδική αναζήτηση στο ευρετήριο παράγει ένα δείκτη στην εγγραφή του αρχείου
- Τα ευρετήρια μπορούν επίσης να χαρακτηρισθούν σαν πυκνά ή αραιά
 - Ένα **πυκνό ευρετήριο** έχει μια καταχώρηση ευρετηρίου για κάθε αναζήτηση τιμής κλειδιού (και επομένως κάθε εγγραφή) στο αρχείο δεδομένων.
 - Ένα **αραιό (ή μη πυκνό) ευρετήριο**, από την άλλη, έχει καταχωρήσεις ευρετηρίου μόνο για κάποιες από τις τιμές αναζήτησης.

Τύποι Ευρετηρίων ενός Επιπέδου

- Πρωτεύον Ευρετήριο
 - Ορίζεται σε ένα ταξινομημένο αρχείο δεδομένων
 - Το αρχείο δεδομένων είναι ταξινομημένο σε ένα **πεδίο κλειδί**
 - Περιλαμβάνει μια καταχώρηση ευρετηρίου για κάθε μπλοκ στο αρχείο δεδομένων· η καταχώρηση ευρετηρίου έχει την τιμή του πεδίου κλειδιού για την πρώτη εγγραφή στο μπλοκ, που ονομάζεται *άγκυρα του μπλοκ*
 - Ένα παρόμοιο σχήμα μπορεί να χρησιμοποιεί την *τελευταία εγγραφή* σε ένα μπλοκ.
 - Ένα πρωτεύον ευρετήριο είναι ένα μη πυκνό (αραιό) ευρετήριο, αφού έχει μια καταχώρηση για κάθε μπλοκ του αρχείου δεδομένων στο δίσκο και στα κλειδιά των εγγραφών άγκυρα παρά για κάθε τιμή αναζήτησης.

Πρωτεύον ευρετήριο στο πεδίο κλειδί ταξινόμησης



Ευρετήρια σαν Μέθοδοι Προσπέλασης (συνέχεια)

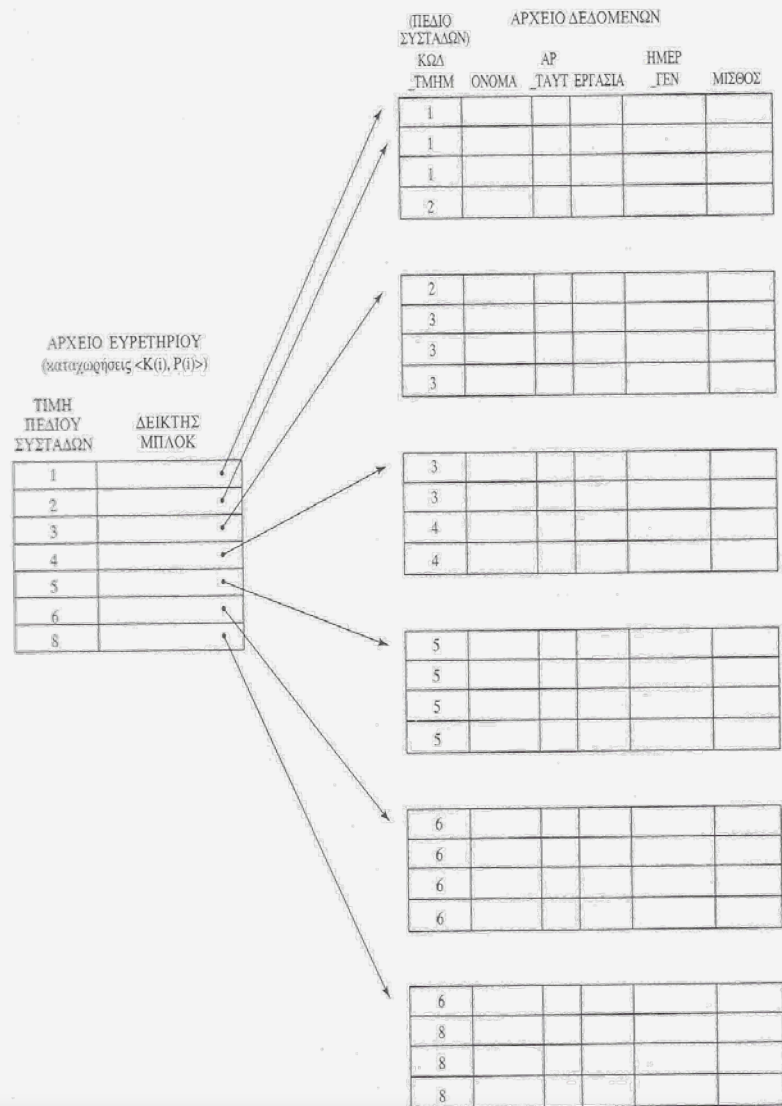
- Παράδειγμα 1: Έστω το αρχείο δεδομένων ΕΡΓΑΖΟΜΕΝΟΣ(ΟΝΟΜΑ, ΑΡ_ΤΑΥΤ, ΔΙΕΥΘΥΝΣΗ, ΕΡΓΑΣΙΑ, ΜΙΣΘΟΣ, ...)
- Υποθέστε ότι:
 - μέγεθος εγγραφής $R=100$ bytes, μέγεθος μπλοκ $B=1024$ bytes, $r=30.000$ εγγραφές
- Τότε, έχουμε:
 - παράγοντας ομαδοποίησης $Bfr = B/R = 1024/100 = 10$ εγγραφές/μπλοκ
 - πλήθος μπλοκ αρχείου $b = (r/Bfr) = (30.000/10) = 3.000$ μπλοκ
 - Κόστος μέσης γραμμικής αναζήτησης: $(b/2) = 3.000/2 = 1.500$ προσπελάσεις μπλοκ
 - Αν οι εγγραφές του αρχείου είναι ταξινομημένες, η δυαδική αναζήτηση είναι:
 $\log_2 b = \log_2 3.000 = 12$ προσπελάσεις μπλοκ
- Για ένα ευρετήριο στο πεδίο ΑΡ_ΤΑΥΤ, υποθέστε μέγεθος πεδίου $V_{ΑΡΤΑΥΤ} = 9$ bytes, υποθέστε μέγεθος δείκτη εγγραφής $P_R = 6$ bytes. Τότε:
 - μέγεθος καταχώρησης ευρετηρίου $R_i = (V_{SSN} + P_R) = (9+6) = 15$ bytes
 - παράγοντας ομαδοποίησης ευρετηρίου $Bfr_i = B/R_i = 1024/15 = 68$ κατχ/μπλοκ
 - πλήθος μπλοκ ευρετηρίου $b_i = (r_i / Bfr_i) = (3.000/68) = 45$ μπλοκ
 - η δυαδική αναζήτηση απαιτεί $\log_2 b_i = \log_2 45 = 6$ μπλοκ προσπελάσεις
 - μία επιπλέον προσπέλαση μπλοκ στο αρχείο δεδομένων $(6+1=7)$

Τύποι Ευρετηρίων ενός Επιπέδου

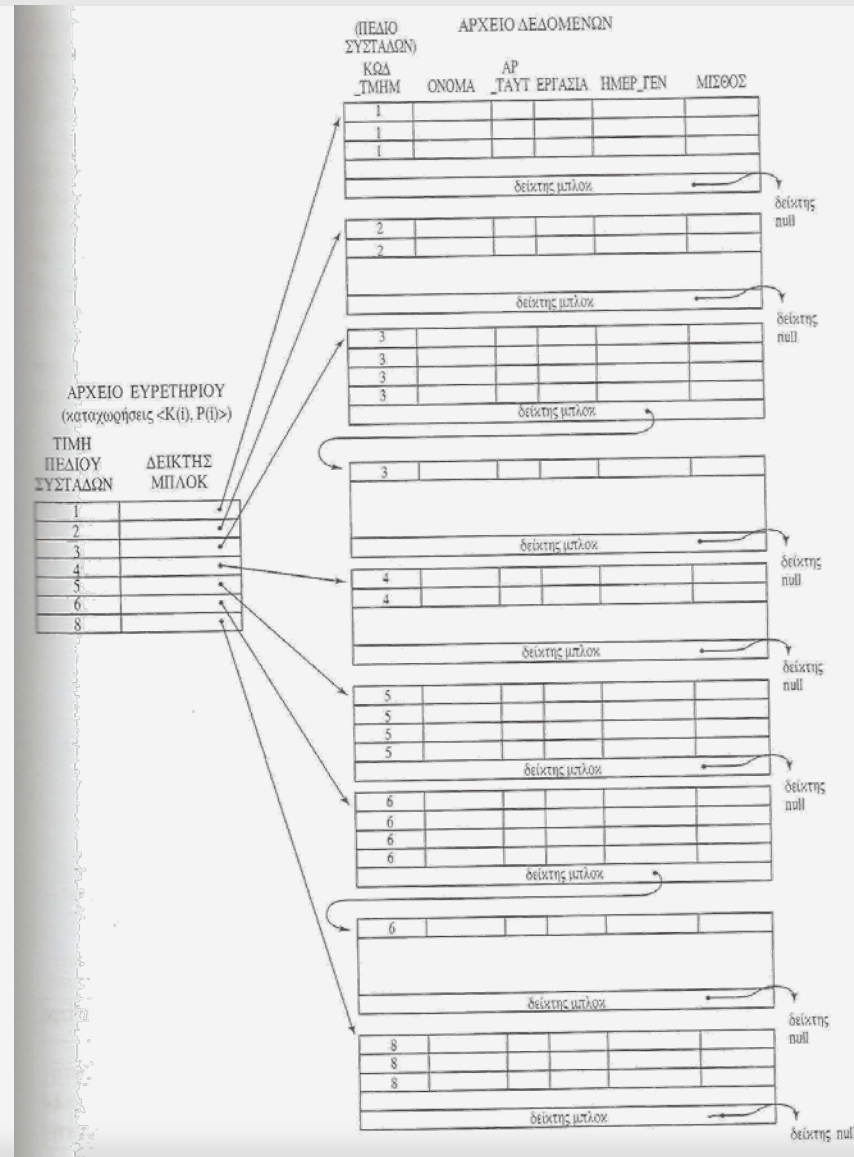
- Ευρετήριο Συστάδα
 - Ορίζεται σε ένα ταξινομημένο αρχείο
 - Το αρχείο δεδομένων είναι ταξινομημένο σε ένα πεδίο που δεν είναι κλειδί, σε αντίθεση από το πρωτεύον ευρετήριο, που απαιτεί ότι το πεδίο ταξινόμησης στο αρχείο δεδομένων έχει μια διακριτή τιμή για κάθε εγγραφή.
 - Περιλαμβάνει μια καταχώρηση ευρετηρίου για κάθε διακριτή τιμή του πεδίου· η καταχώρηση ευρετηρίου δείχνει στο πρώτο μπλοκ δεδομένων που περιέχει εγγραφές με αυτή την τιμή πεδίου.
 - Είναι ένα ακόμη παράδειγμα μη πυκνού ευρετηρίου όπου η εισαγωγή και η διαγραφή είναι σχετικά εύκολη με ένα ευρετήριο συστάδα.

Ένα παράδειγμα ευρετηρίου συστάδας

- Ένα ευρετήριο συστάδα στο πεδίο ταξινόμησης ΚΩΔ_ΤΜΗΜ που δεν είναι κλειδί στο αρχείο ΕΡΓΑΖΟΜΕΝΟΣ.



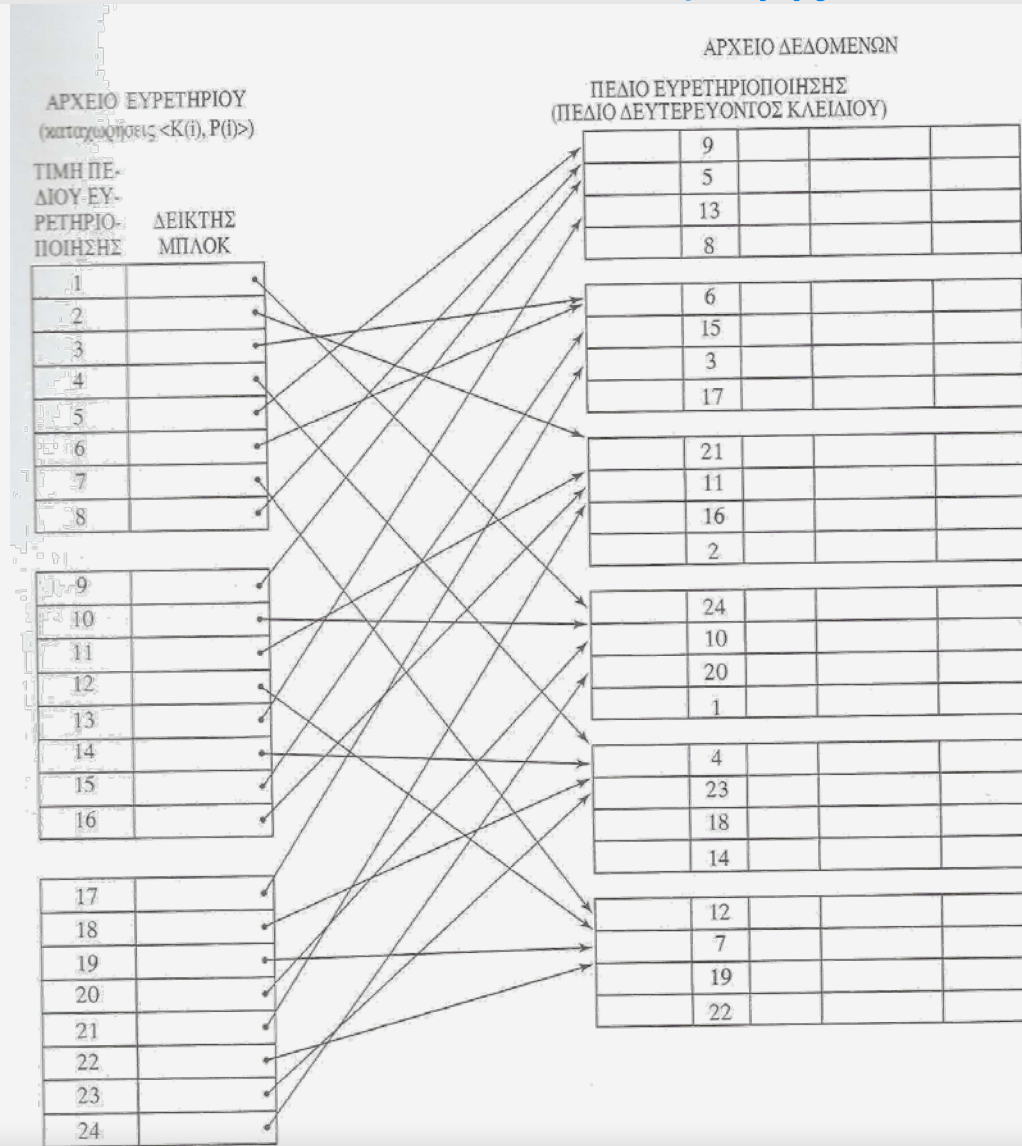
Ένα ακόμη παράδειγμα ευρετηρίου συστάδας



Τύποι Ευρετηρίων ενός Επιπέδου

- Δευτερεύον Ευρετήριο
 - Ένα δευτερεύον ευρετήριο υποστηρίζει ένα δευτερεύοντα τρόπο προσπέλασης ενός αρχείου για το οποίο υπάρχει ήδη πρωτεύουσα οργάνωση.
 - Το δευτερεύον ευρετήριο μπορεί να είναι σε ένα πεδίο που είναι υποψήφιο κλειδί και έχει μοναδική τιμή σε κάθε εγγραφή, ή ένα πεδίο που δεν είναι κλειδί με διπλές τιμές.
 - Το ευρετήριο είναι ένα ταξινομημένο αρχείο με δύο πεδία.
 - Το πρώτο πεδίο είναι ίδιου τύπου δεδομένων με κάποιο πεδίο που δεν είναι κλειδί του αρχείου δεδομένων και ονομάζεται πεδίο ευρετηρίου.
 - Το δεύτερο πεδίο είναι δείκτης **μπλοκ** ή δείκτης εγγραφής.
 - Μπορεί να υπάρχουν πολλά δευτερεύοντα ευρετήρια (και επομένως, πεδία ευρετηρίασης) για το ίδιο αρχείο.
 - Περιλαμβάνει μια καταχώρηση για *κάθε εγγραφή* στο αρχείο δεδομένων· επομένως, είναι ένα *πυκνό ευρετήριο*.

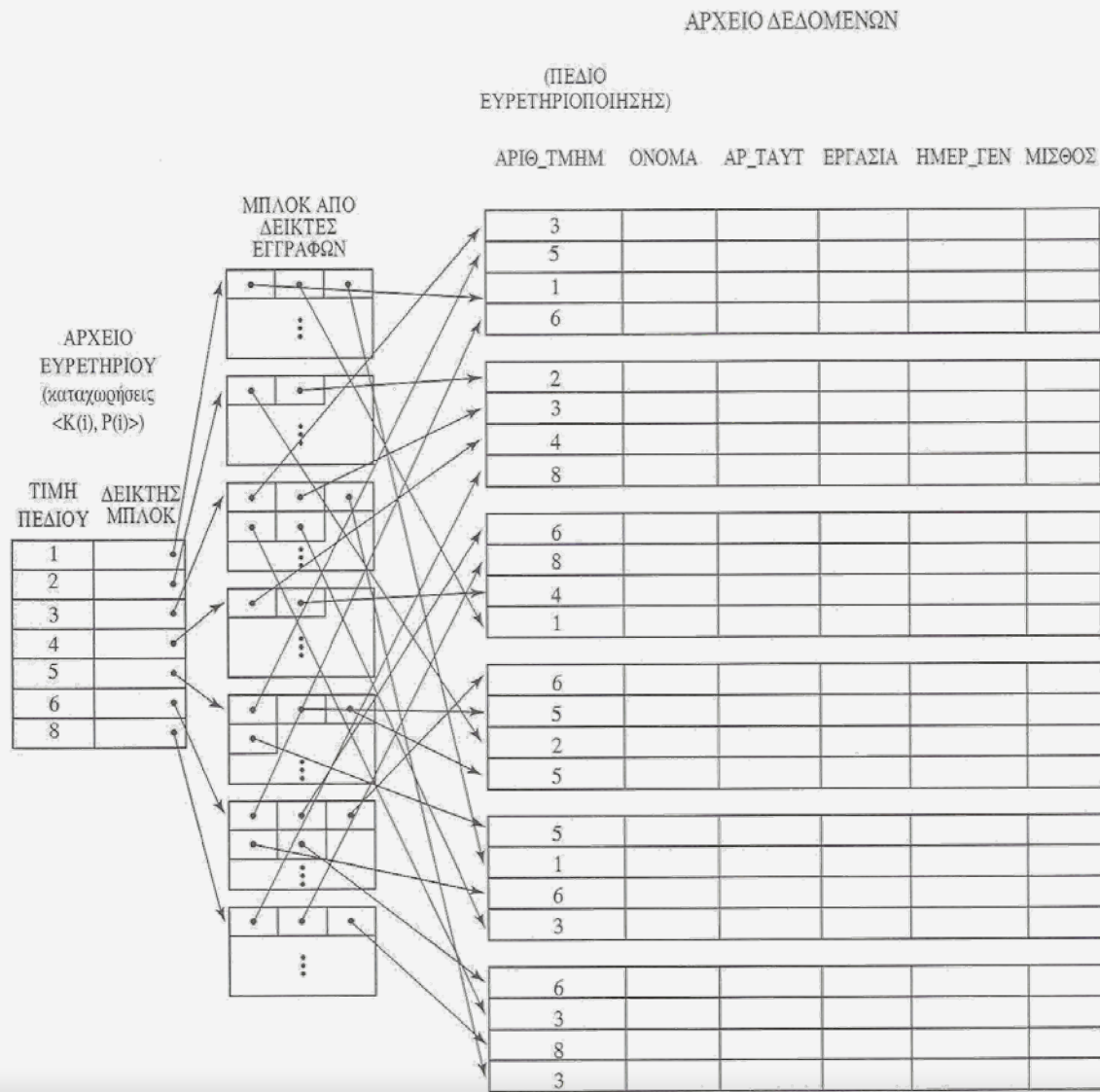
Πυκνό Δευτερεύον Ευρετήριο προς κάποιο μη διατεταγμένο πεδίο κλειδιού ενός αρχείου



Ευρετήρια σαν Μέθοδοι Προσπέλασης

- Παράδειγμα 2: Έστω το αρχείο δεδομένων ΕΡΓΑΖΟΜΕΝΟΣ(ΟΝΟΜΑ, ΑΡ_ΤΑΥΤ, ΔΙΕΥΘΥΝΣΗ, ΕΡΓΑΣΙΑ, ΜΙΣΘΟΣ, ...)
- Υποθέστε ότι:
 - μέγεθος εγγραφής $R=100$ bytes, μέγεθος μπλοκ $B=1024$ bytes, $r=30.000$ εγγραφές
- Τότε, έχουμε:
 - παράγοντας ομαδοποίησης $Bfr = B/R = 1024/100 = 10$ εγγραφές/μπλοκ
 - πλήθος μπλοκ αρχείου $b = (r/Bfr) = (30.000/10) = 3.000$ μπλοκ
 - Κόστος μέσης γραμμικής αναζήτησης: $(b/2) = 3.000/2 = 1.500$ προσπελάσεις μπλοκ
- Για ένα δευτερεύων ευρετήριο στο πεδίο ΜΙΣΘΟΣ, υποθέστε μέγεθος πεδίου $V_{\text{ΜΙΣΘΟΣ}} = 9$ bytes, υποθέστε μέγεθος δείκτη εγγραφής $P_R = 6$ bytes. Τότε:
 - μέγεθος καταχώρησης ευρετηρίου $R_i = (V_{\text{SSN}} + P_R) = (9+6) = 15$ bytes
 - παράγοντας ομαδοποίησης ευρετηρίου $Bfr_i = B/R_i = 1024/15 = 68$ κατχ/μπλοκ
 - πλήθος μπλοκ ευρετηρίου $b_i = (r_i / Bfr_i) = (30.000/68) = 442$ μπλοκ
 - η δυαδική αναζήτηση απαιτεί $\log_2 b_i = \log_2 442 = 9$ μπλοκ προσπελάσεις
 - μία επιπλέον προσπέλαση μπλοκ στο αρχείο δεδομένων ($9+1=10$)

Δευτερεύον Ευρετήριο ως προς πεδίο που δεν είναι κλειδί



Τύποι Ευρετηρίων

	Πεδίο Ευρετηρίου που χρησιμοποιείται για τη φυσική διάταξη του αρχείου	Το πεδίο ευρετηρίου δεν χρησιμοποιείται για τη φυσική διάταξη του αρχείου
Πεδίο Κλειδί	Πρωτεύον Ευρετήριο	Δευτερεύον Ευρετήριο (κλειδί)
Πεδίο μη κλειδί	Ευρετήριο Συστάδων	Δευτερεύον Ευρετήριο (μη κλειδί)

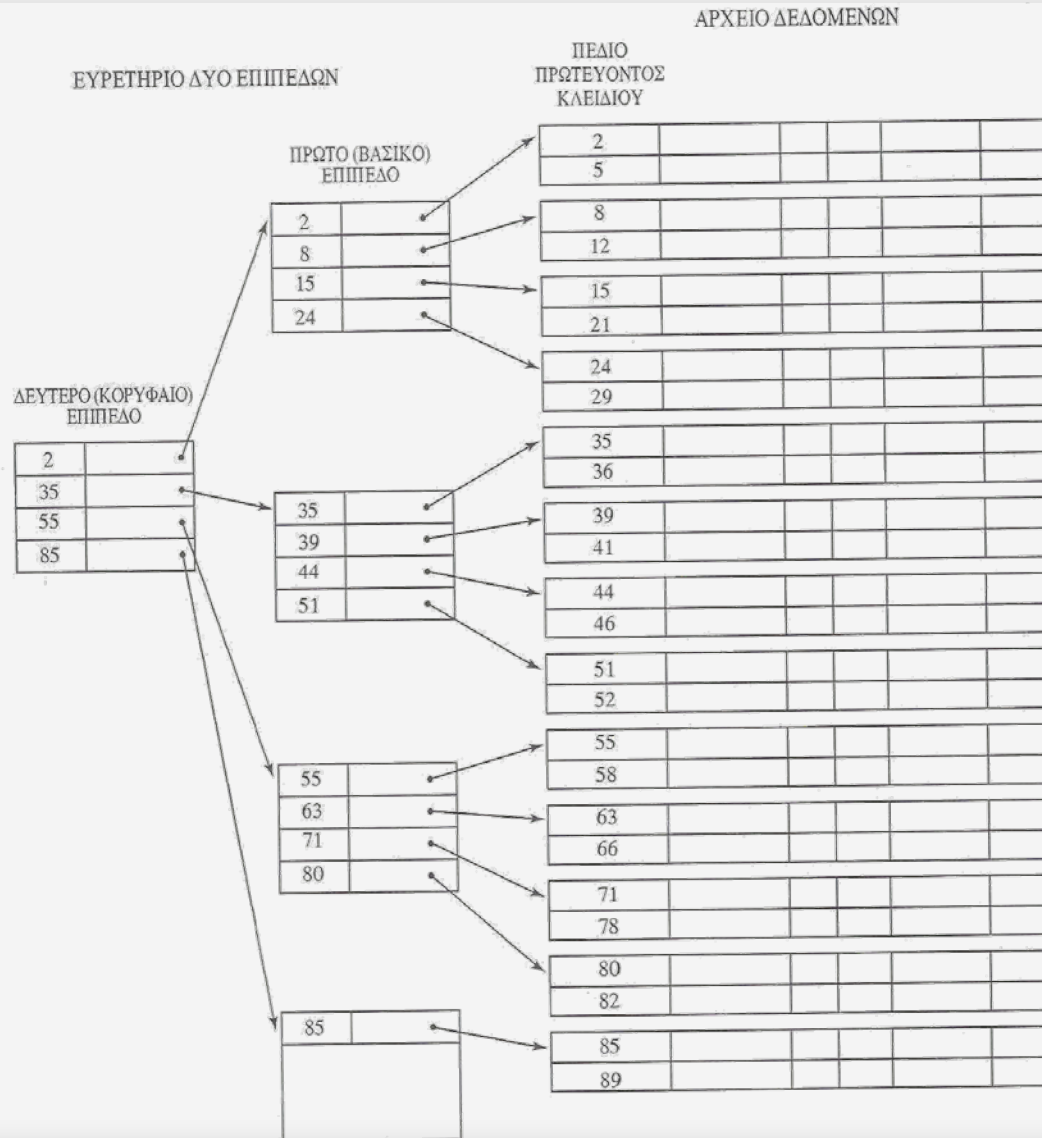
Ιδιότητες Τύπων Ευρετηρίων

Τύπος Ευρετηρίου	Πλήθος Καταχωρήσεων του Ευρετηρίου (Πρώτου επιπέδου)	Πυκνό ή Μη Πυκνό	Άγκυρες μπλοκ στο Αρχείο Δεδομένων
Πρωτεύον Ευρετήριο	Πλήθος μπλοκ του αρχείου δεδομένων	Μη Πυκνό	Ναι
Ευρετήριο Συστάδων	Πλήθος διαφορετικών τιμών του πεδίου ευρετηρίασης	Μη Πυκνό	Ναι/Όχι
Δευτερεύον Ευρετήριο πάνω σε πεδίο-κλειδί	Πλήθος εγγραφών αρχείου δεδομένων	Πυκνό	Όχι
Δευτερεύον Ευρετήριο πάνω σε πεδίο-μη κλειδί	Πλήθος εγγραφών ή πλήθος διαφορετικών τιμών του πεδίου ευρετηριοποίησης	Πυκνό ή Μη Πυκνό	Όχι

Ευρετήρια Πολλών Επιπέδων

- Επειδή ένα ευρετήριο ενός επιπέδου είναι ένα ταξινομημένο αρχείο, μπορούμε να δημιουργήσουμε ένα πρωτεύον ευρετήριο στο ίδιο το ευρετήριο.
- Στην περίπτωση αυτή, το αρχικό αρχείο ευρετηρίου ονομάζεται *πρώτο επίπεδο ευρετηρίου* και το ευρετήριο του ευρετηρίου ονομάζεται *δεύτερο επίπεδο ευρετηρίου*.
- Μπορούμε να επαναλάβουμε τη διαδικασία αυτή, δημιουργώντας ένα τρίτο, τέταρτο ... επίπεδο, μέχρι που όλες οι καταχωρήσεις στο *ψηλότερο επίπεδο* να χωράνε σε ένα μπλοκ μνήμης.
- Ένα ευρετήριο πολλών επιπέδων μπορεί να δημιουργηθεί οποιοδήποτε τύπο ευρετηρίου πρώτου επιπέδου (πρωτεύον, δευτερεύον, συστάδα) όσο το πρώτο επίπεδο ευρετηρίου αποτελείται από περισσότερα από ένα μπλοκ δίσκου

Πρωτεύον Ευρετήριο δύο επιπέδων



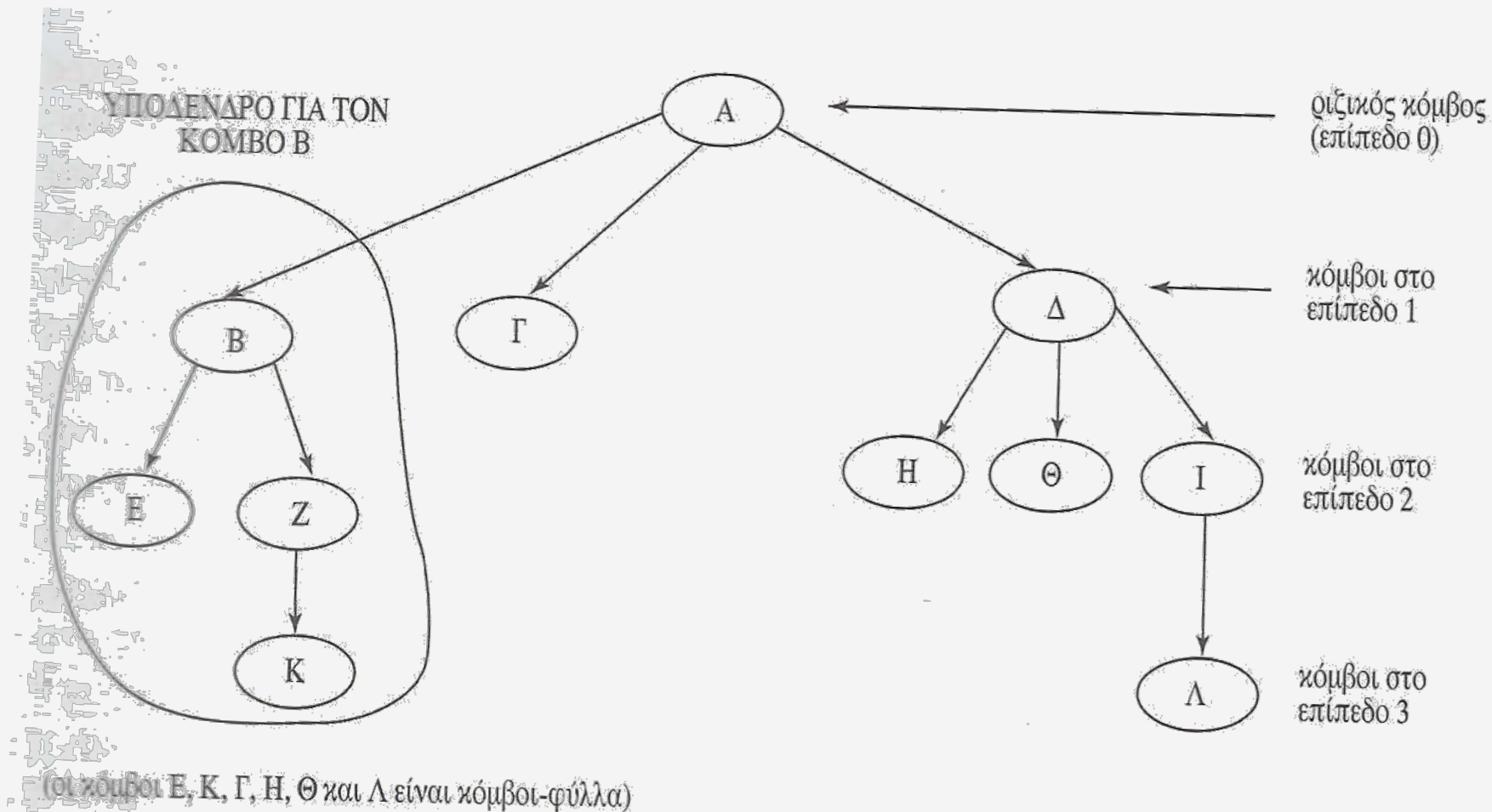
Ευρετήρια σαν Μέθοδοι Προσπέλασης

- Παράδειγμα 3: Έστω το αρχείο δεδομένων ΕΡΓΑΖΟΜΕΝΟΣ(ΟΝΟΜΑ, ΑΡ_ΤΑΥΤ, ΔΙΕΥΘΥΝΣΗ, ΕΡΓΑΣΙΑ, ΜΙΣΘΟΣ, ...)
- Υποθέστε ότι:
 - μέγεθος εγγραφής $R=100$ bytes, μέγεθος μπλοκ $B=1024$ bytes, $r=30.000$ εγγραφές
- Για ένα δευτερεύων ευρετήριο στο ένα πεδίο, υποθέστε μέγεθος πεδίου $V_{ΑΡΤΑΥΤ}=9$ bytes, υποθέστε μέγεθος δείκτη εγγραφής $P_R=6$ bytes. Τότε:
 - μέγεθος καταχώρησης ευρετηρίου $R_i=(V_{SSN}+P_R)=(9+6)=15$ bytes
 - παράγοντας ομαδοποίησης ευρετηρίου $Bfr_i=1024/15=68$ κατχ/μπλοκ
 - πλήθος μπλοκ ευρετηρίου $b_1=(r_1/Bfr_i)=(30.000/68)=442$ μπλοκ
 - πλήθος μπλοκ ευρετηρίου $b_2=(b_1/Bfr_i)=(442/68)=7$ μπλοκ
 - πλήθος μπλοκ ευρετηρίου $b_3=(b_2/Bfr_i)=(7/68)=1$ μπλοκ
 - Προσπέλαση ενός μπλοκ από κάθε επίπεδο συν ένα μπλοκ από το αρχείο δεδομένων ($3+1=4$ προσπελάσεις μπλοκ)

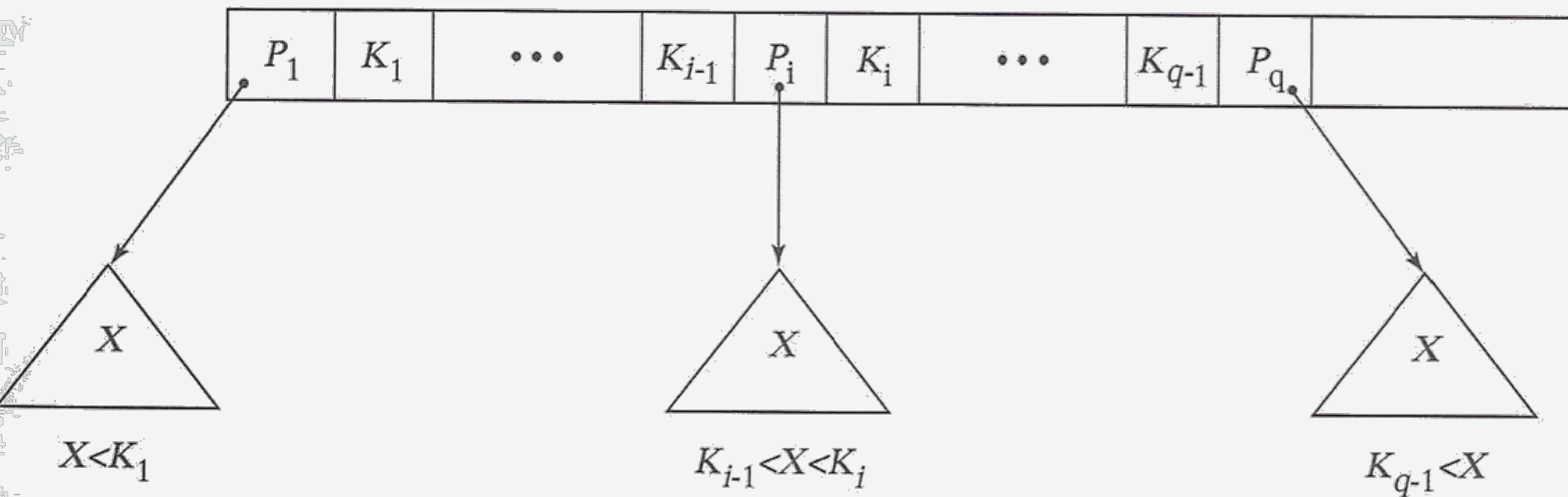
Ευρετήρια Πολλών επιπέδων

- Ένα τέτοιο ευρετήριο πολλών επιπέδων είναι μια μορφή *δένδρου αναζήτησης*
- Ωστόσο, η εισαγωγή και η διαγραφή καταχωρήσεων στο ευρετήριο αποτελεί σοβαρό πρόβλημα επειδή κάθε επίπεδο του ευρετηρίου είναι ένα *ταξινομημένο αρχείο*.

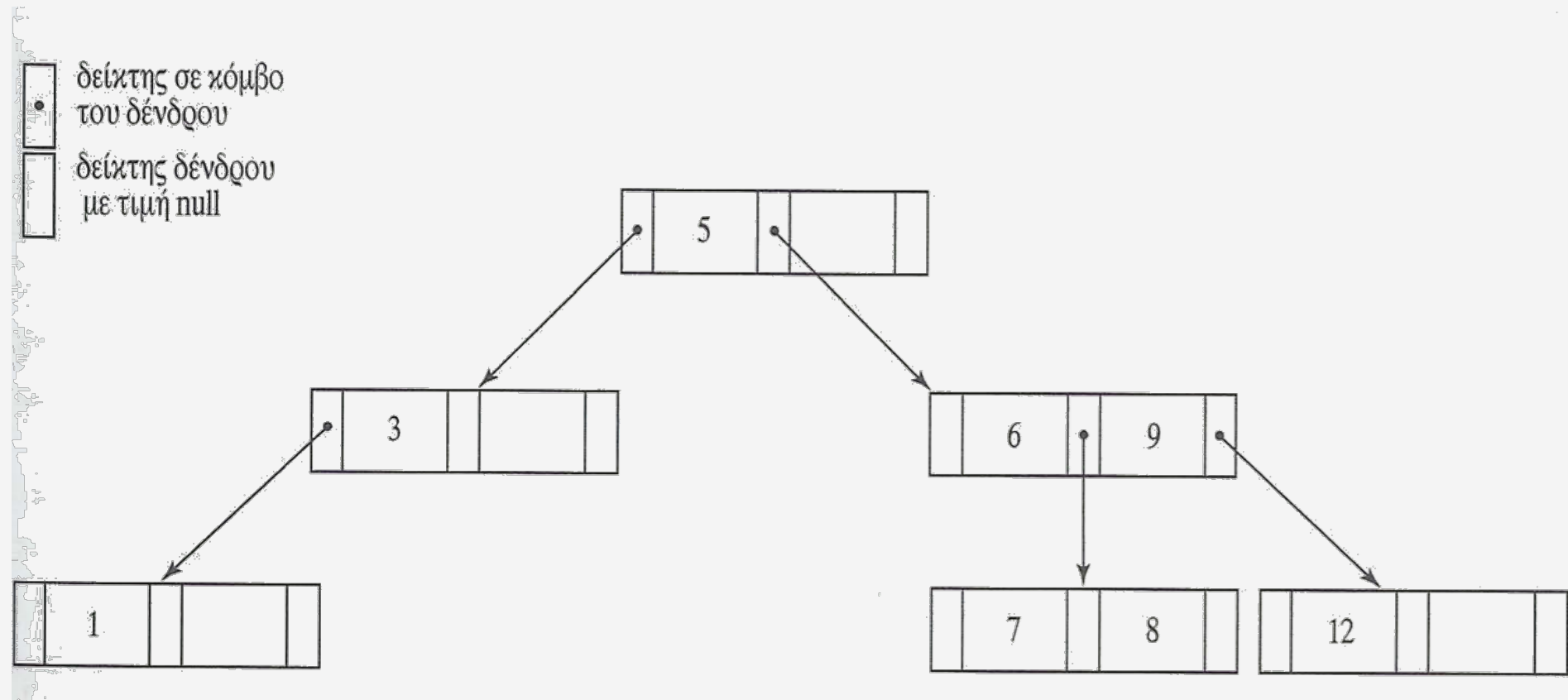
Δενδρική Δομή Δεδομένων



Ένας κόμβος σε ένα δέντρο αναζήτησης με δείκτες στα υποδέντρα του



Ένας κόμβος σε ένα δέντρο αναζήτησης τάξεως $p=3$



Δυναμικά Πολυεπίπεδα Ευρετήρια με Χρήση B-Δένδρων και B+-Δένδρων

- Τα περισσότερα ευρετήρια πολλών επιπέδων χρησιμοποιούν δομές δεδομένων B-δένδρων ή B+-δένδρων λόγω του προβλήματος εισαγωγής και διαγραφής
 - Αυτό αφήνει σε κάθε κόμβο του δένδρου (μπλοκ στο δίσκο) ελεύθερο χώρο για νέες καταχωρήσεις
- Αυτές οι δομές δεδομένων αποτελούν παραλλαγές των δένδρων αναζήτησης που υποστηρίζουν αποτελεσματική εισαγωγή και διαγραφή νέων τιμών αναζήτησης αναζήτησης.
- Στις δομές δεδομένων B-δένδρων και B+-δένδρων, κάθε κόμβος αντιστοιχεί σε ένα μπλοκ του δίσκου
- Κάθε κόμβος είναι από γεμάτος ή μέχρι τη μέση

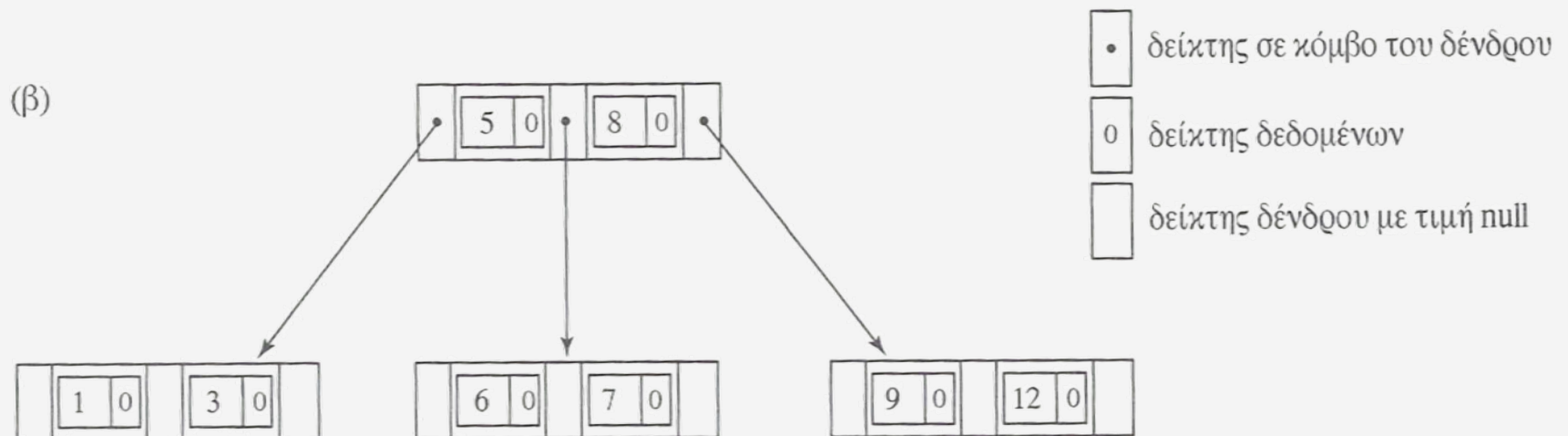
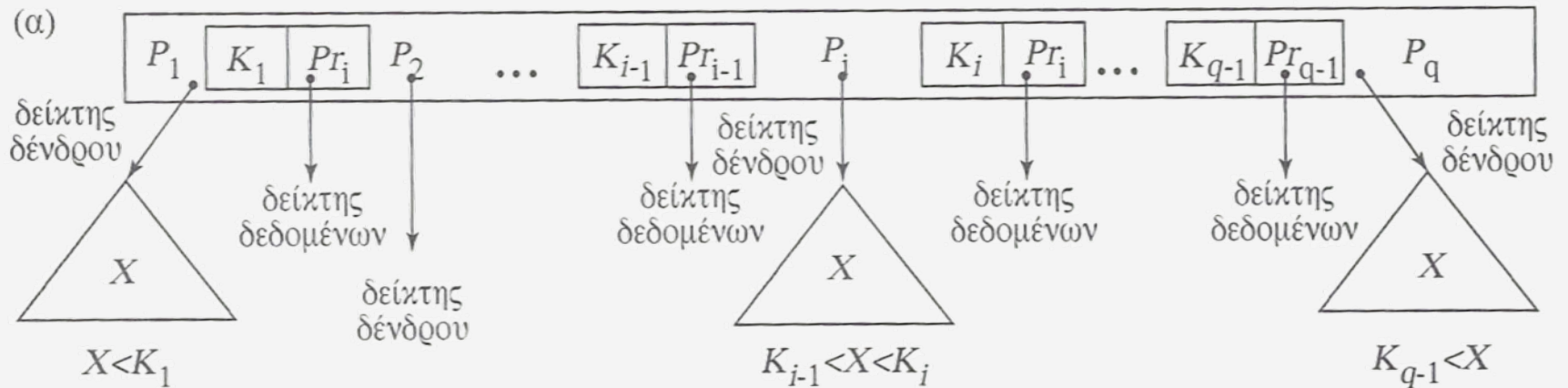
Δυναμικά Πολυεπίπεδα Ευρετήρια με Χρήση B-Δένδρων και B+-Δένδρων (συνέχεια)

- Η εισαγωγή σε ένα κόμβο που δεν είναι γεμάτος είναι πολύ αποδοτική
 - Αν ένας κόμβος είναι γεμάτος η εισαγωγή προκαλεί διάσπαση σε δύο κόμβους
- Η διάσπαση μπορεί να διαδοθεί σε άλλα επίπεδα του δένδρου
- Μια διαγραφή είναι πολύ αποδοτική αν ένας κόμβος δεν πέφτει κάτω από το μισό
- Αν μια διαγραφή οδηγήσει σε κόμβο με λιγότερες από μισές καταχωρήσεις, πρέπει να συνενωθεί με γειτονικούς κόμβους

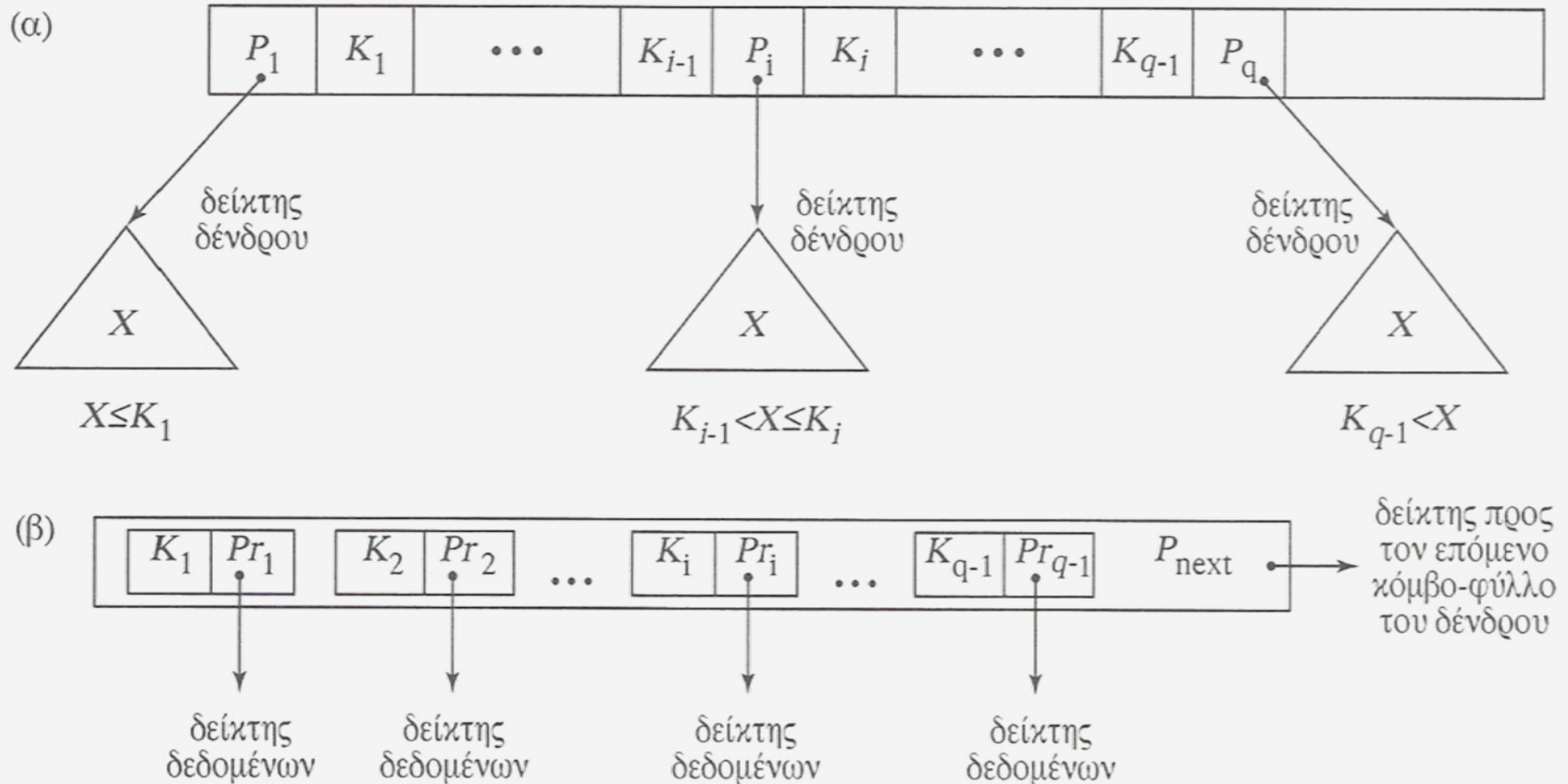
Διαφορά μεταξύ B-δένδρου και B⁺-δένδρου

- Σε ένα B-δένδρου, υπάρχουν δείκτες προς εγγραφές δεδομένων σε όλα τα επίπεδα του δένδρου
- Σε ένα B⁺-δένδρο, όλοι οι δείκτες προς τις εγγραφές δεδομένων υπάρχουν στους κόμβους φύλα
- Ένα B⁺-δένδρο μπορεί να έχει λιγότερα επίπεδα (ή μεγαλύτερη χωρητικότητα τιμών αναζήτησης) από το αντίστοιχο B-δένδρο

Δομές B-δέντρου



Οι κόμβοι ενός B⁺-δέντρου



- (α) Εσωτερικός κόμβος ενός B⁺-δέντρου με $q-1$ τιμές αναζήτησης
(β) Κόμβος-φύλλο ενός B⁺-δέντρου

Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

- ΑΚΟΛΟΥΘΙΑ ΕΙΣΑΓΩΓΗΣ: 8, 5, 1, 7, 3, 12, 9, 6



Δείκτης προς κόμβο δένδρου



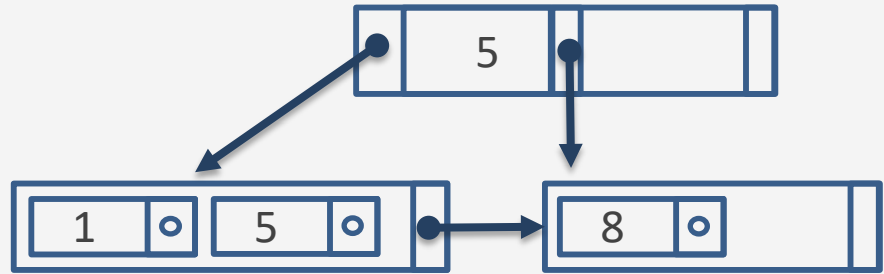
Δείκτης δεδομένων



Δείκτης δένδρου με τιμή null



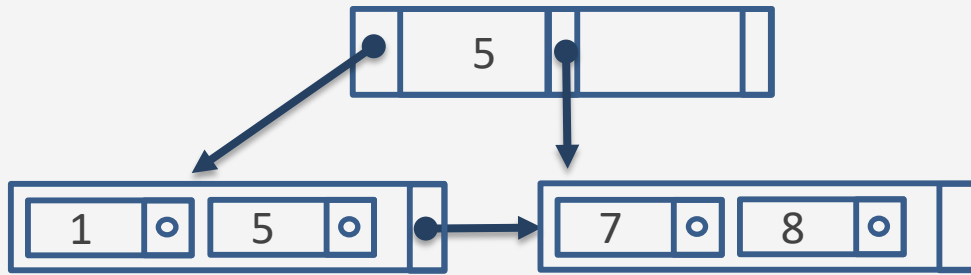
Εισαγωγή του 1:
υπερχείλιση (νέο επίπεδο)



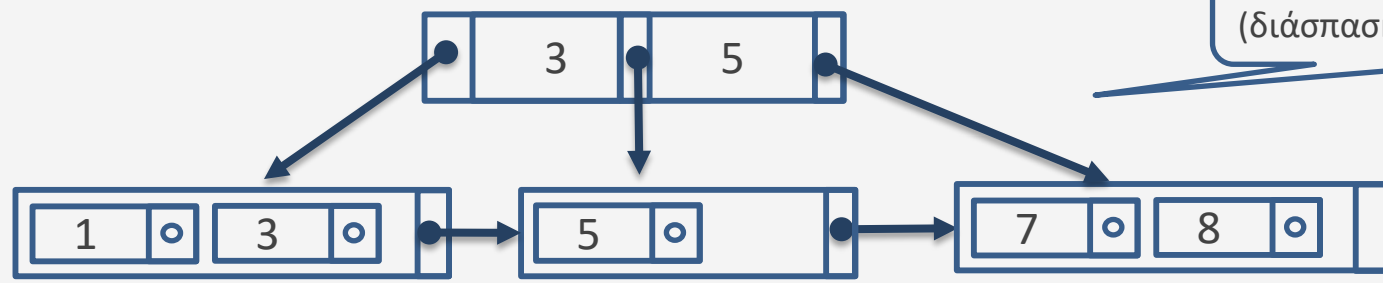
Εισαγωγή του 7

Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

- ΑΚΟΛΟΥΘΙΑ ΕΙΣΑΓΩΓΗΣ: 8, 5, 1, 7, 3, 12, 9, 6



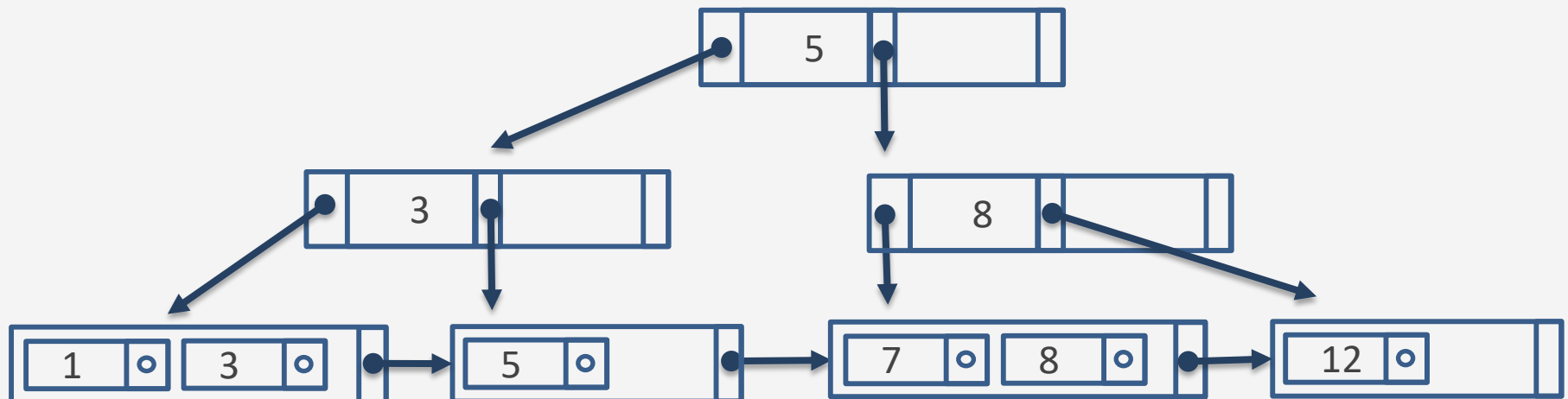
Εισαγωγή του 3:
υπερχείλιση (διάσπαση)



Εισαγωγή του 12: υπερχείλιση
(διάσπαση, διαδίδεται, νέο επίπεδο)

Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

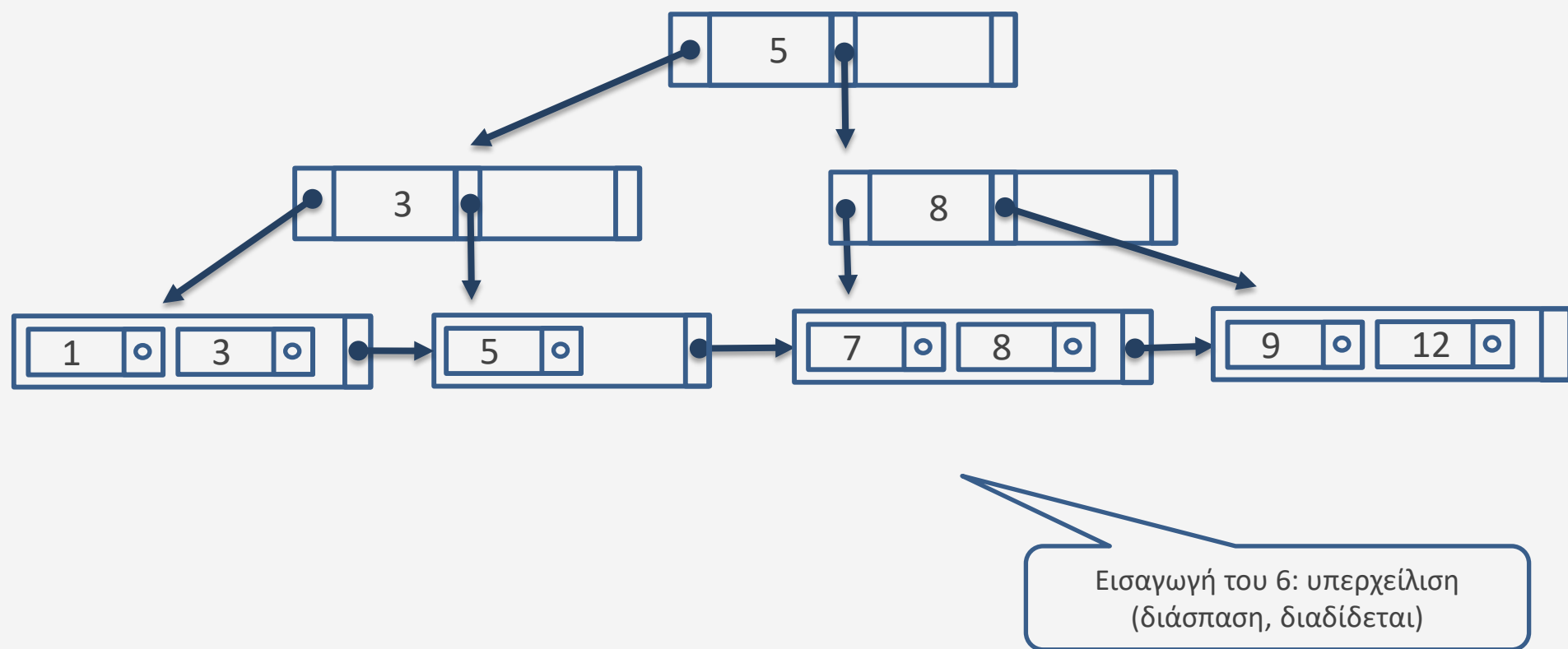
- ΑΚΟΛΟΥΘΙΑ ΕΙΣΑΓΩΓΗΣ: 8, 5, 1, 7, 3, 12, 9, 6



Εισαγωγή του 9

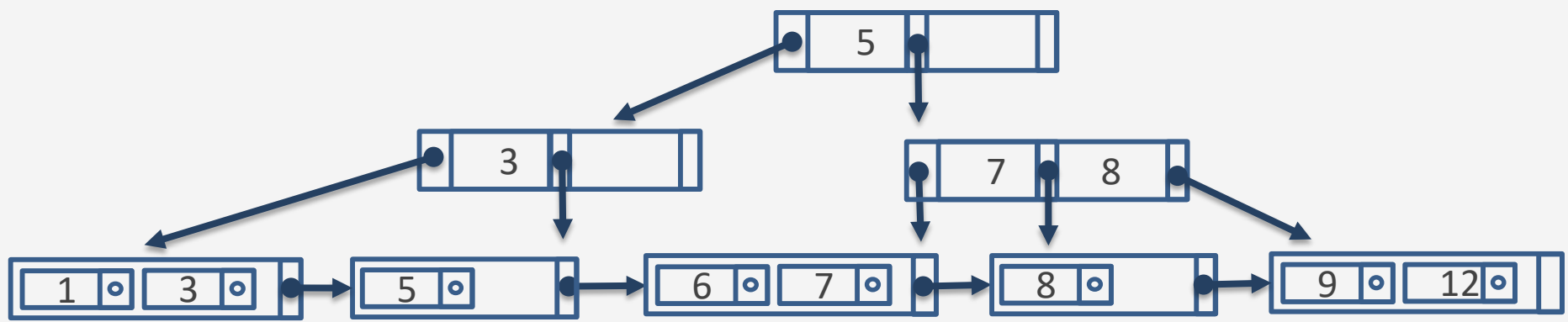
Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

- ΑΚΟΛΟΥΘΙΑ ΕΙΣΑΓΩΓΗΣ: 8, 5, 1, 7, 3, 12, 9, 6



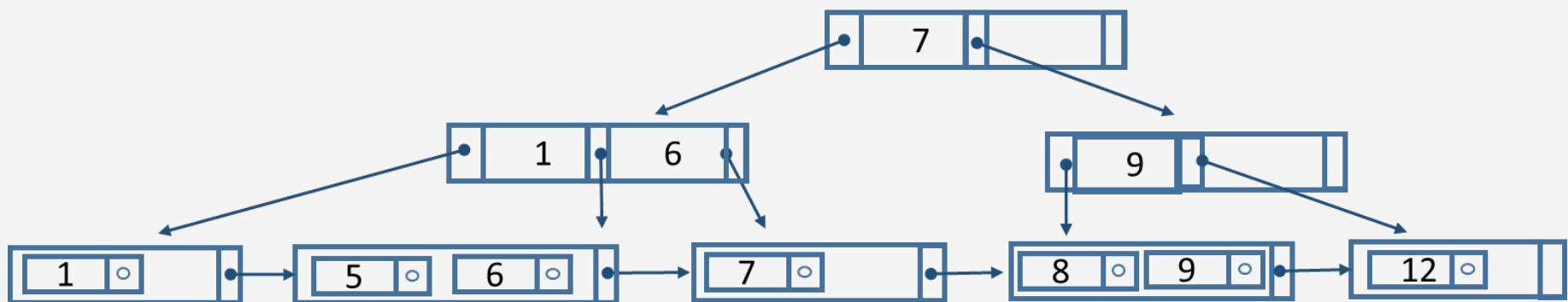
Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

- ΑΚΟΛΟΥΘΙΑ ΕΙΣΑΓΩΓΗΣ: 8, 5, 1, 7, 3, 12, 9, 6



Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

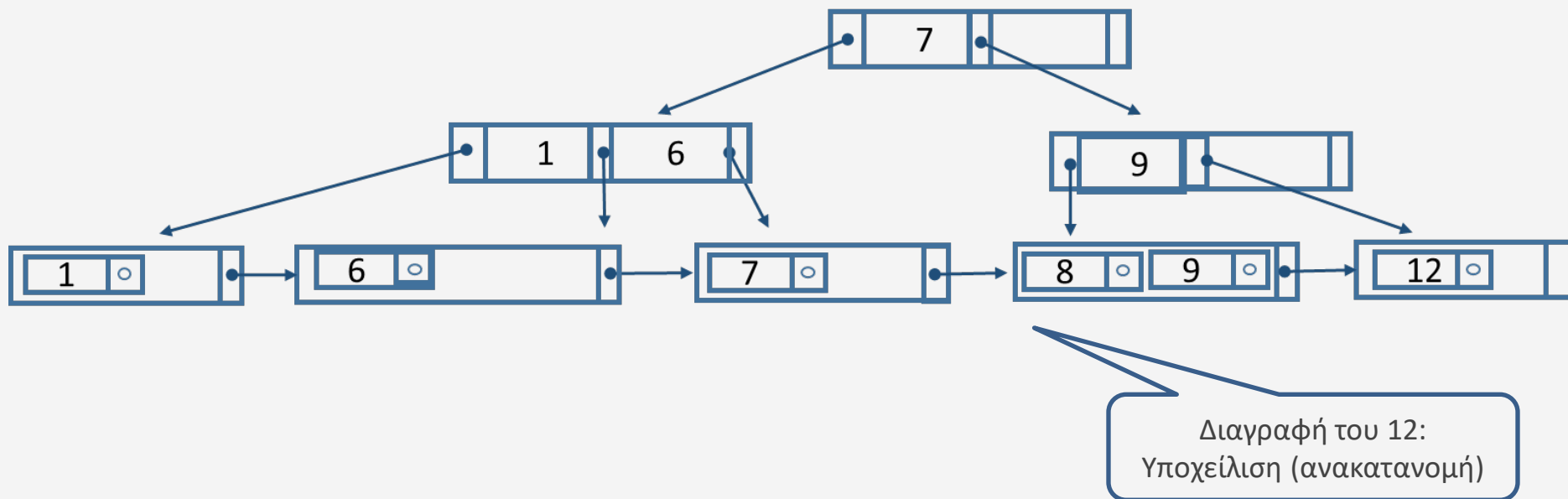
- ΑΚΟΛΟΥΘΙΑ ΔΙΑΓΡΑΦΗΣ: 5, 12, 9



Διαγραφή του 5

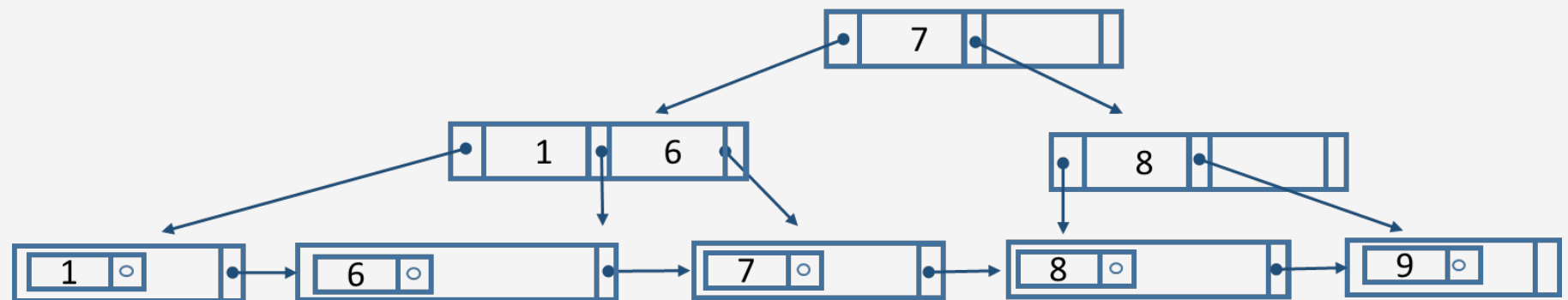
Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

- ΑΚΟΛΟΥΘΙΑ ΔΙΑΓΡΑΦΗΣ: 5, 12, 9



Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

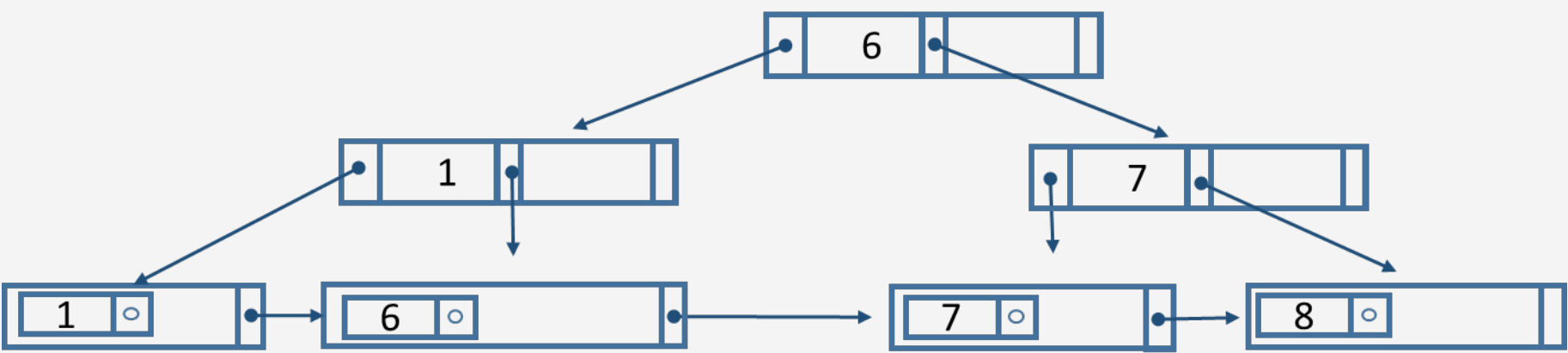
- ΑΚΟΛΟΥΘΙΑ ΔΙΑΓΡΑΦΗΣ: 5, 12, 9



Διαγραφή του 9:
Υποχείλιση (συγχώνευση, με αριστερό
κόμβο, και πάλι υποχείλιση, συνένωση
επιπέδων)

Ένα παράδειγμα εισαγωγής σε B⁺-δέντρο

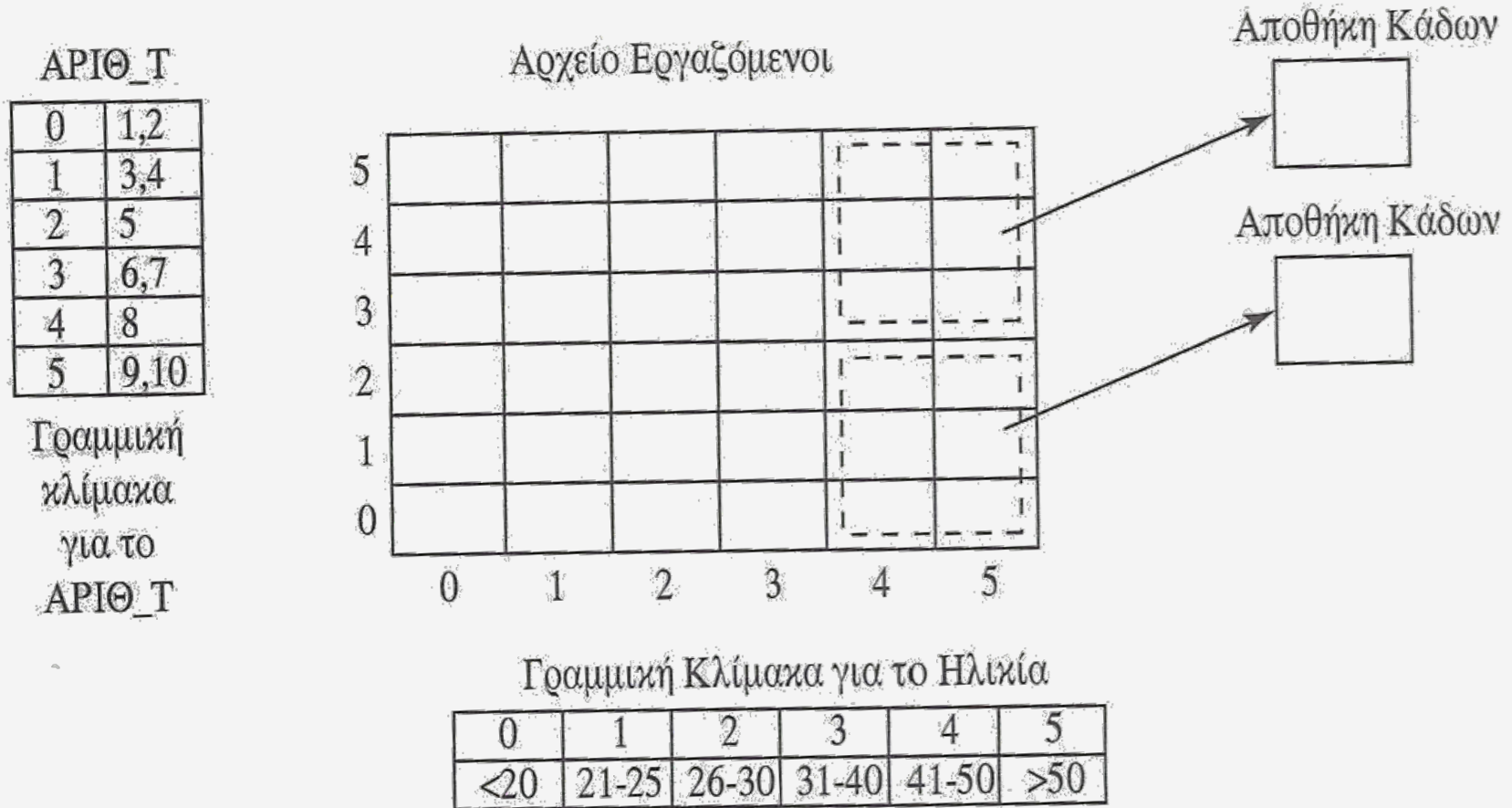
- ΑΚΟΛΟΥΘΙΑ ΔΙΑΓΡΑΦΗΣ: 5, 12, 9



Ευρετήρια σε πολλαπλά κλειδιά

- Δομή προσπέλασης πολλαπλών γνωρισμάτων
- Σύνθετα κλειδιά:
 - Κλειδιά που περιλαμβάνουν πολλαπλά γνωρίσματα
- Ταξινομημένα ευρετήριο σε πολλαπλά γνωρίσματα
- Διαμερισμένος κατακερματισμός
- Δικτυωτά Αρχεία (grid files)

Παράδειγμα δικτυωτού αρχείου



Ευρετηρίαση πολυδιάστατων δεδομένων

- Είναι το σχεσιακό μοντέλο των ΣΔΒΔ κατάλληλο για τα πολυδιάστατα δεδομένα (π.χ. χωρικά δεδομένα);
 - Τα σχεσιακά ΣΔΒΔ παρέχουν απλούς τύπους τιμών, όπως αριθμοί και συμβολοσειρές
 - Έννοια της ολικής διάταξης
- Τα δένδρα αναζήτησης είναι αποτελεσματικά για αριθμούς
 - Οι έννοιες αυτές δεν έχουν ιδιαίτερο νόημα για τα χωρικά δεδομένα (π.χ. σημεία στο επίπεδο)

Ευρετηρίαση πολυδιάστατων δεδομένων (συνέχεια)

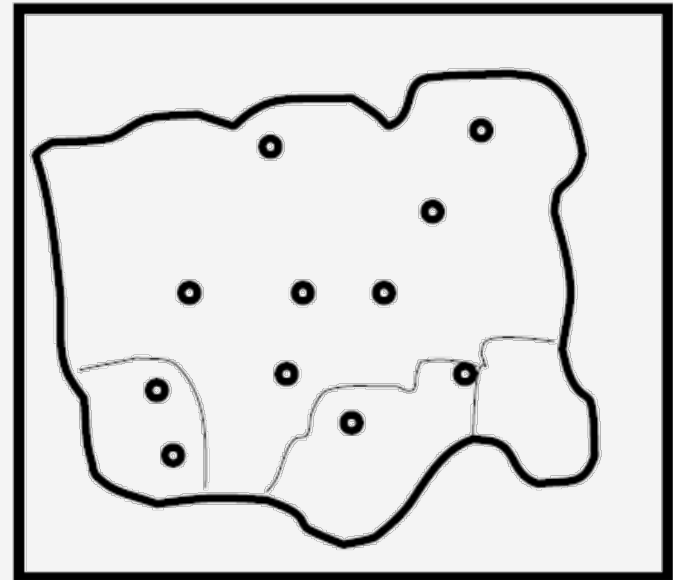
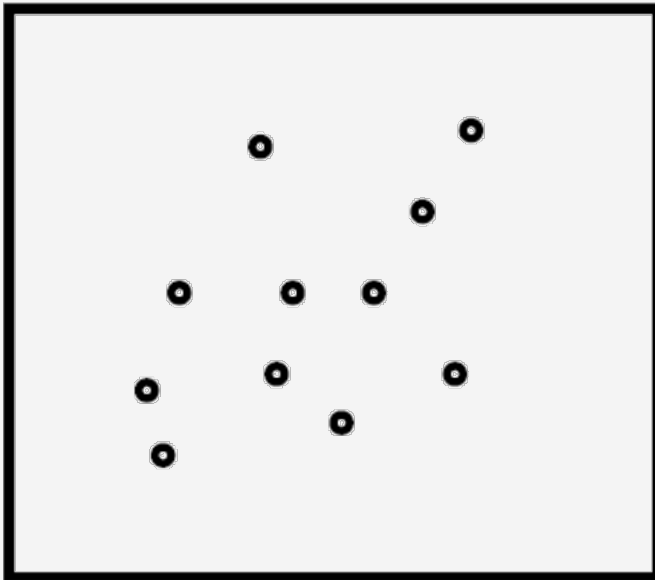
- Εφαρμογές GIS (χάρτες):
 - Πολεοδομικός σχεδιασμός, βελτιστοποίηση διαδρομής, έλεγχος φωτιάς ή ρύπανσης, κ.λπ.
- Άλλες εφαρμογές:
 - Σχεδιασμός VLSI, CAD/CAM, μοντελοποίηση ανθρώπινου εγκεφάλου, κλπ..
- Παραδοσιακές εφαρμογές:
 - Πολυμεσικές εγγραφές

Ευρετηρίαση πολυδιάστατων δεδομένων (συνέχεια)

παραδοσιακές DB

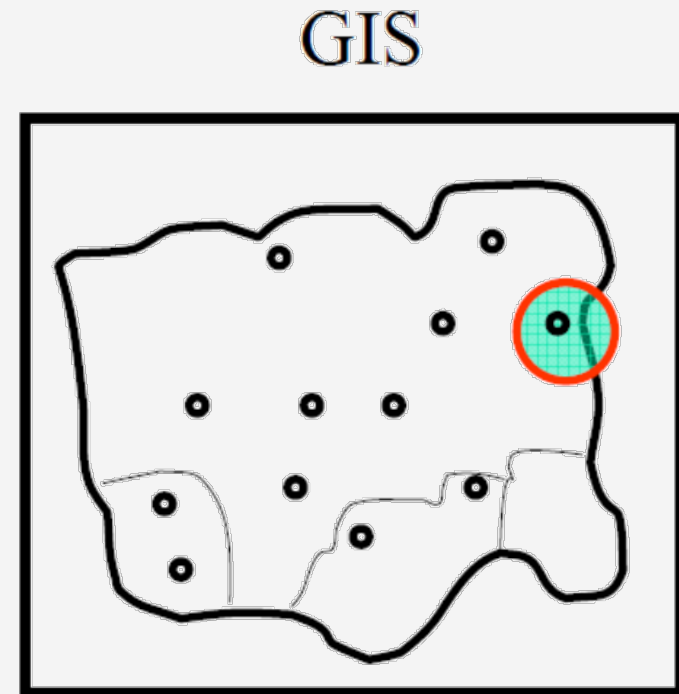
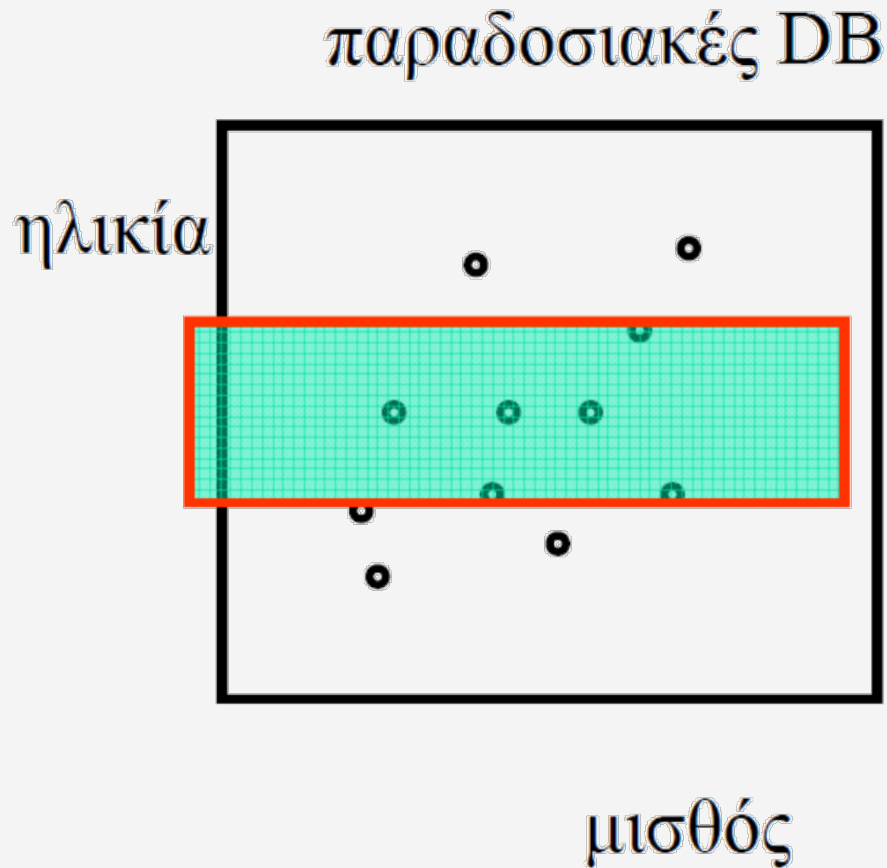
GIS

ηλικία



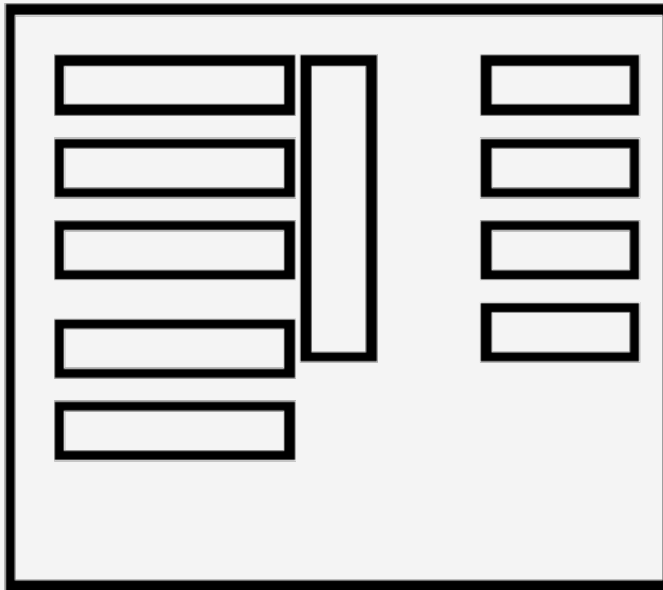
μισθός

Ευρετηρίαση πολυδιάστατων δεδομένων (συνέχεια)



Ευρετηρίαση πολυδιάστατων δεδομένων (συνέχεια)

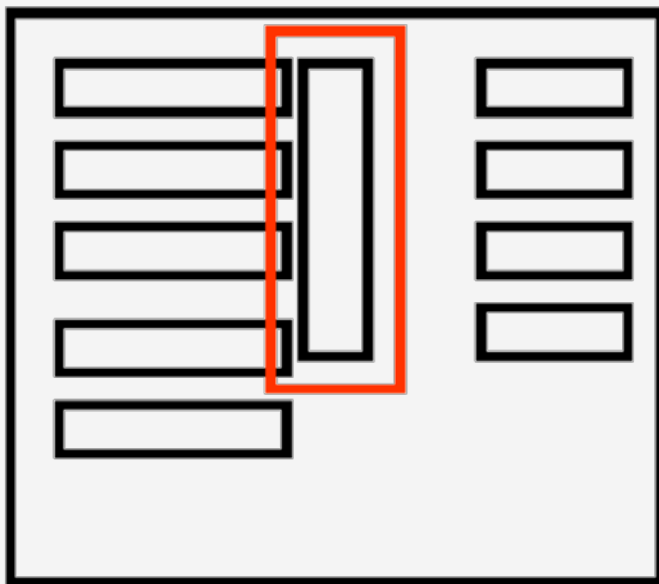
CAD/CAM



βρείτε στοιχεία
πολύ κοντά
το ένα στο άλλο

Ευρετηρίαση πολυδιάστατων δεδομένων (συνέχεια)

CAD/CAM

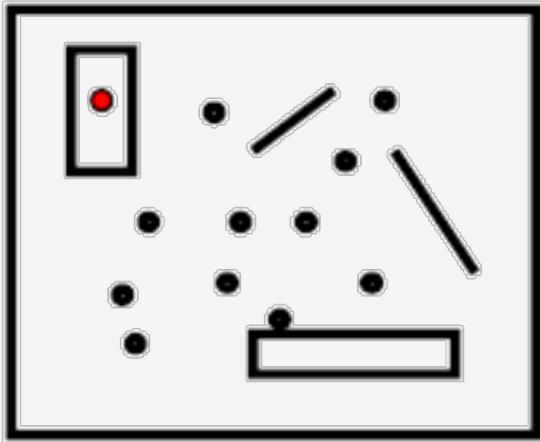


Απλές χωρικές ερωτήσεις και λειτουργίες

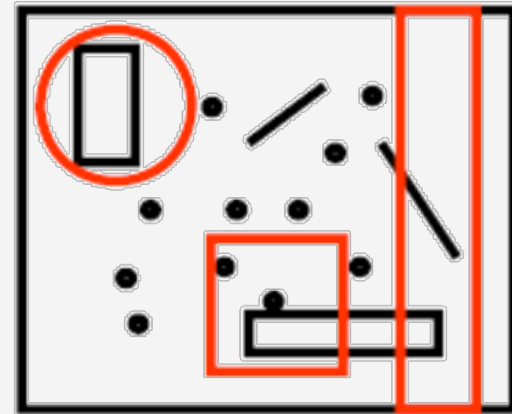
- Το φυσικό μοντέλο παρέχει απλές λειτουργίες που χρειάζονται στην επίλυση χωρικών ερωτήσεων.
- Συνηθισμένες ερωτήσεις:
 - **Point query:** Βρες όλα τα αντικείμενα που περικλείουν ένα δοθέν σημείο
 - **Range query:** Βρες όλα τα αντικείμενα που βρίσκονται μέσα σε μια (συνήθως ορθογώνια) περιοχή
 - **Nearest neighbor query:** Βρες το πλησιέστερο αντικείμενο σε σχέση με ένα σημείο
 - **Spatial Join** (μεταξύ δύο συνόλων αντικειμένων): Βρες όλα τα ζεύγη αντικειμένων (ένα από κάθε σύνολο) που ικανοποιούν μια χωρική συνθήκη (συνήθως, overlap)

Παραδείγματα χωρικών ερωτήσεων

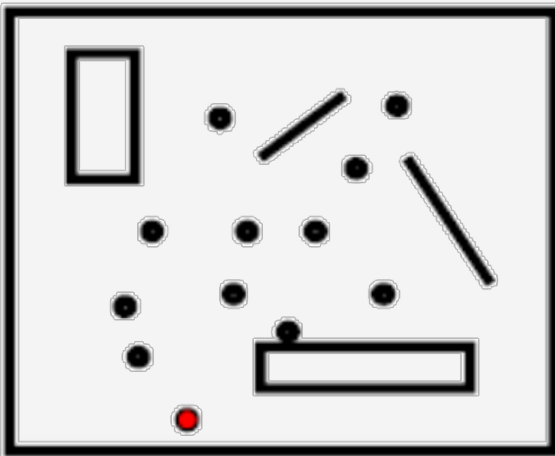
Point query



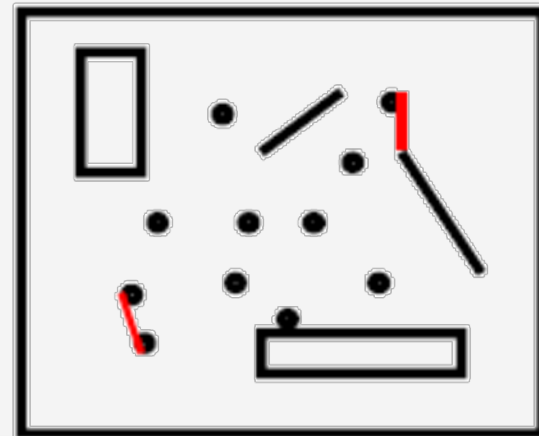
Range query



Nearest neighbor query

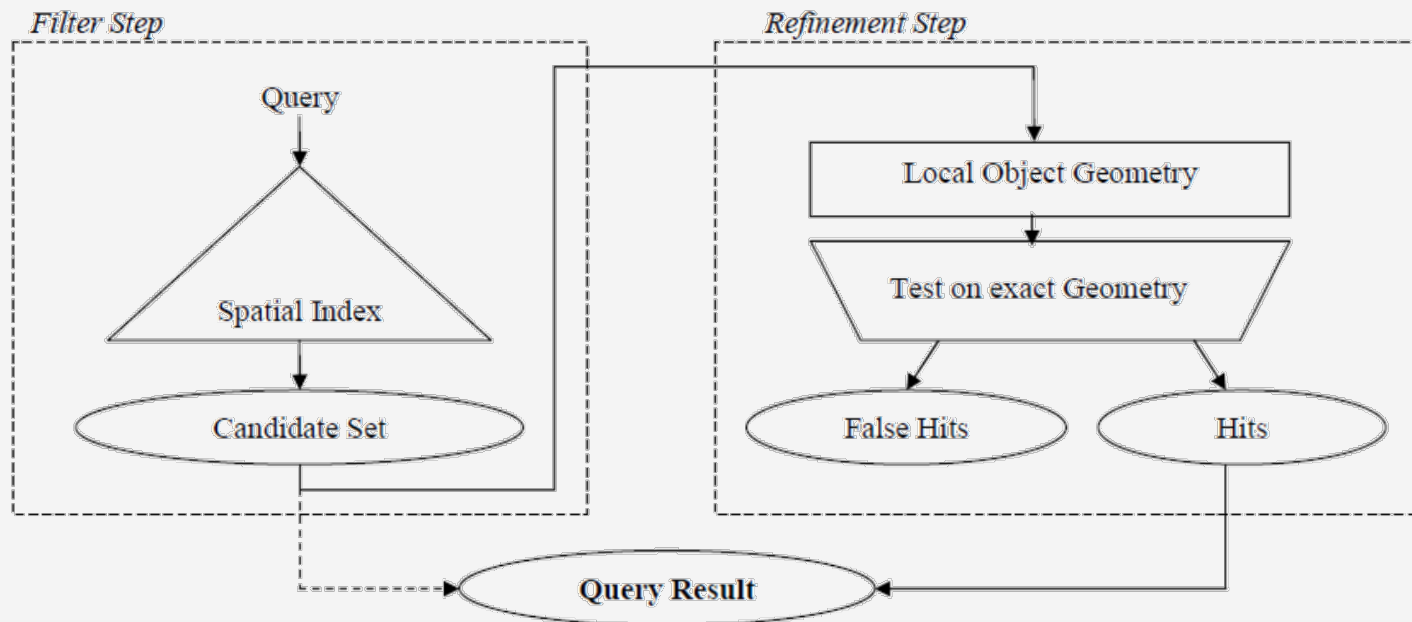


Spatial Join



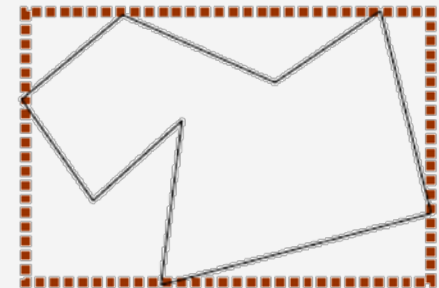
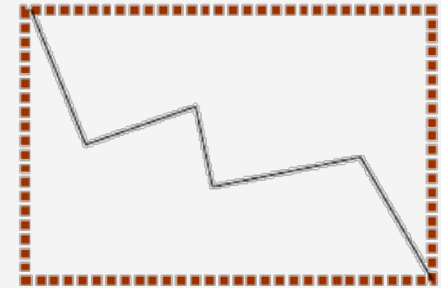
Η διαδικασία «Filter-Refinement»

- Επεξεργασία μίας χωρικής ερώτησης Q
 - **βήμα φιλτραρίσματος (Filter)**: βρες ένα σύνολο S που σίγουρα περιέχει **και** τις απαντήσεις του Q
 - Με χρήση προσεγγιστικών χωρικών τύπων και λειτουργιών
 - **βήμα εκλέπτυνσης (Refinement)**: βρες τις ακριβείς απαντήσεις του Q κάνοντας χρήση ενός GIS για την επεξεργασία του S
 - Με χρήση επακριβών χωρικών τύπων και λειτουργιών

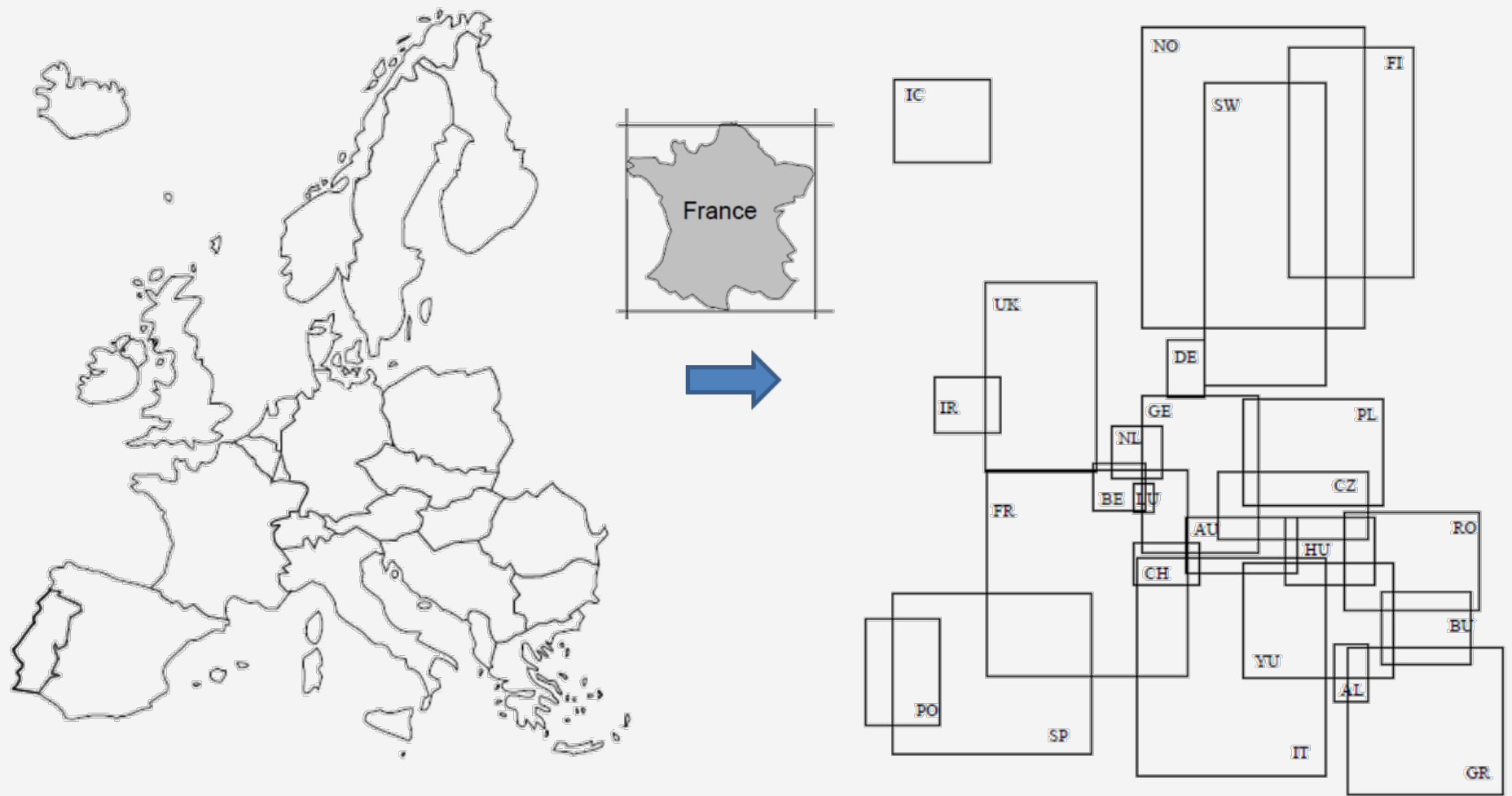


Προσεγγιστικοί χωρικοί τύποι

- **Minimum Bounding Rectangle (MBR)**
 - προσεγγίζει γραμμές, πολύγωνα, ...
 - $MBR(obj)$ είναι το μικρότερο ορθογώνιο παραλληλόγραμμο (παραλληλεπίπεδο κλπ.) που περιβάλλει το αντικείμενο obj και είναι παράλληλο με το καρτεσιανό σύστημα συντεταγμένων
 - Τα MBRs χρησιμοποιούνται κατά κόρον από χωρικά ευρετήρια, π.χ. R-tree
 - Οι αλγόριθμοι για χωρικές λειτουργίες σε MBRs είναι απλοί
 - Τα Χωρικά ΣΔΒΔ επεξεργάζονται MBRs στο βήμα filter (για λόγους επίδοσης)

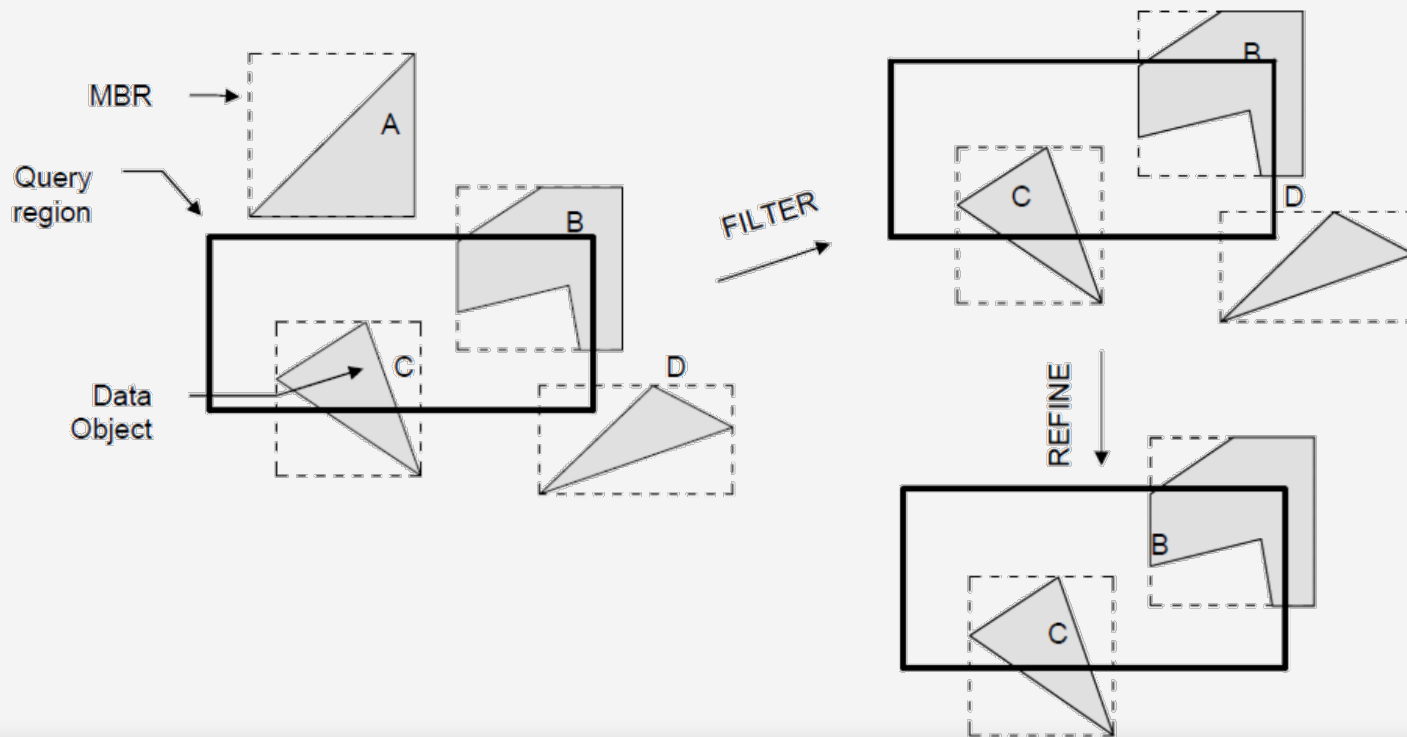


Παράδειγμα MBR



Παράδειγμα filter – refinement

- Range queries
 - στο filter βήμα εκτελείται ο (προσεγγιστικός και πιο φθηνός) τελεστής $\text{overlap}(\text{MBR}(A), Q)$
 - στο refinement βήμα εκτελείται ο (ακριβής αλλά και χρονοβόρος) τελεστής $\text{overlap}(A, Q)$



Οργάνωση αρχείων & ευρετήρια

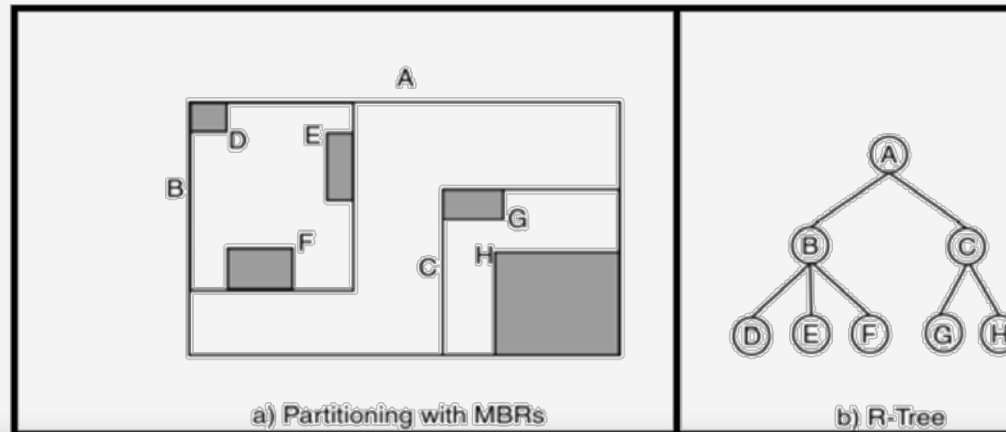
- Επαναχρησιμοποίηση εννοιών σχεσιακού φυσικού μοντέλου – η έννοια της διάταξης (ordering)
 - Οι **καμπύλες διάσχισης χώρου** (space filling curves) ορίζουν μία ολική διάταξη για τα σημεία
 - Αυτή η ολική διάταξη βοηθάει στη χρήση διατεταγμένων αρχείων (ordered files) ή δένδρων αναζήτησης (search trees)
 - π.χ. B-trees
 - Αλλά μπορεί να οδηγήσει σε αναποτελεσματικούς υπολογισμούς!
 - Καινοτόμες τεχνικές
 - Χωρικά ευρετήρια, π.χ. δένδρα περιοχών (R-trees), δένδρα τεταρτημορίων (Quadtree), αρχεία πλέγματος (Gridfile)
 - Παρέχουν καλύτερες αποδόσεις στους υπολογισμούς

Ευρετηρίαση χωρικών δεδομένων

- Τα MBRs χρησιμοποιούνται κατά κόρον από χωρικά ευρετήρια, π.χ. R-δένδρο
- Το R-δένδρο είναι επέκταση του B^+ -δένδρου στις k διαστάσεις, όπου $k > 1$.
- Στις δύο διαστάσεις τα χωρικά αντικείμενα στο R-δένδρο προσεγγίζονται από το ελάχιστο καλύπτον ορθογώνιο (minimum boundary rectangle –MBR).

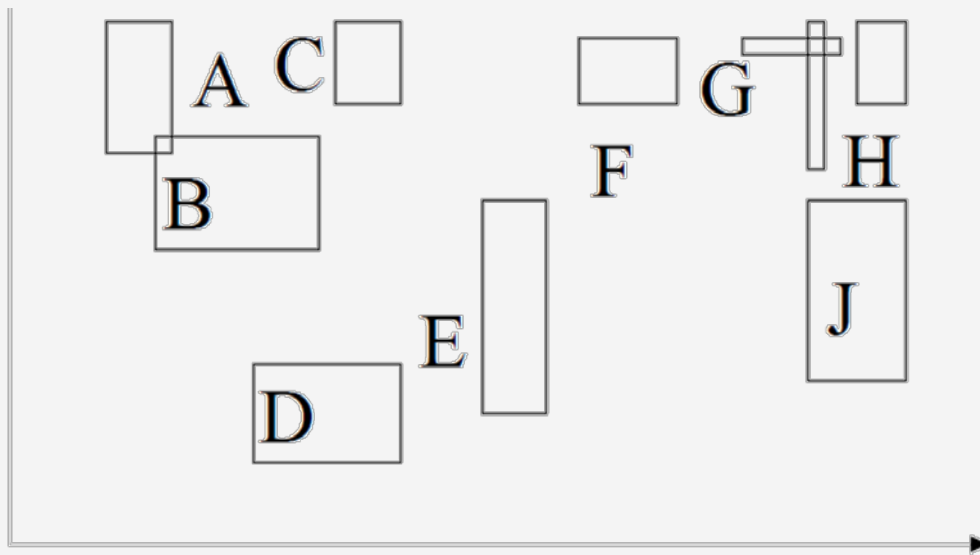
R-δένδρο

- Βασική ιδέα
 - Γενίκευση B-tree σε χωρικά δεδομένα
- Ιδιότητες των R-trees
 - Κάθε κόμβος αντιστοιχεί σε μία σελίδα του δίσκου
 - Ισοζυγισμένα / Οι κόμβοι είναι ορθογώνια / Το ορθογώνιο-παιδί περιέχεται πλήρως εντός του ορθογωνίου-πατέρα
 - Μπορεί να προκύψει χωρική επικάλυψη μεταξύ των κόμβων
 - Η αναζήτηση γίνεται με κατάβαση στο δένδρο



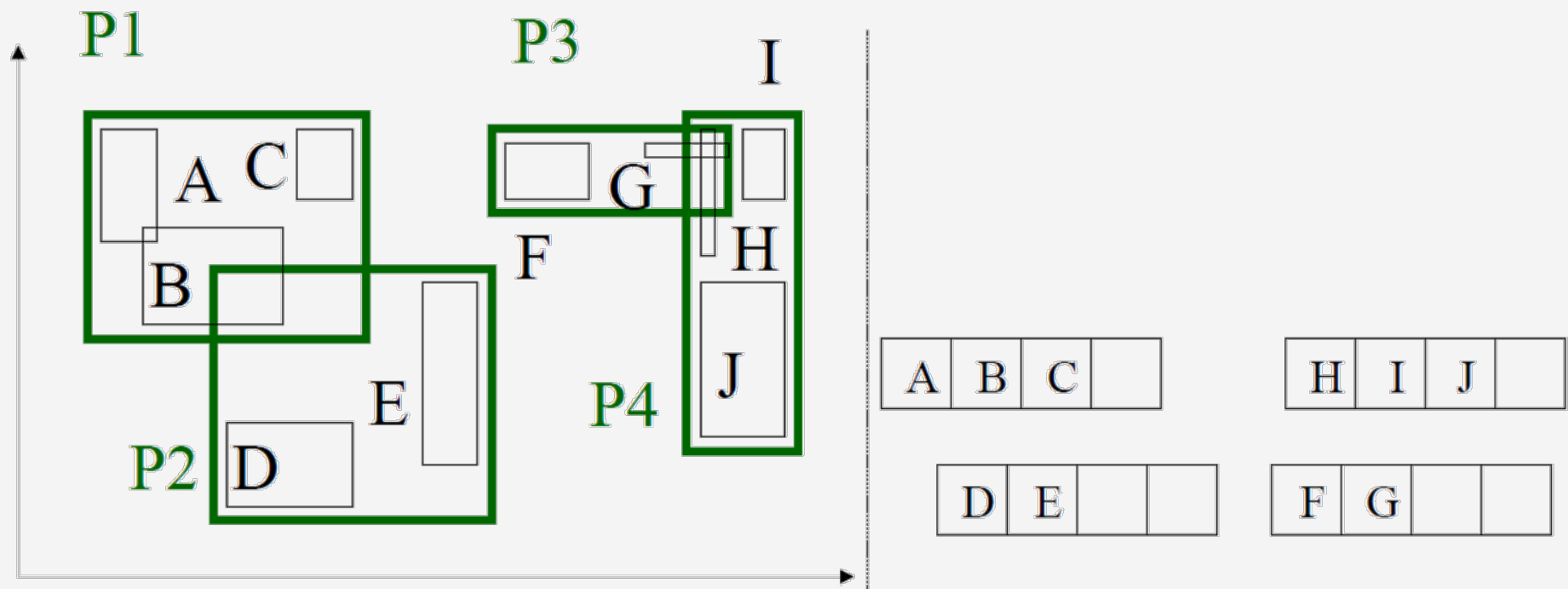
R-δένδρα

- Ομαδοποίησε κοντινά ορθογώνια στους γονείς MBRs.

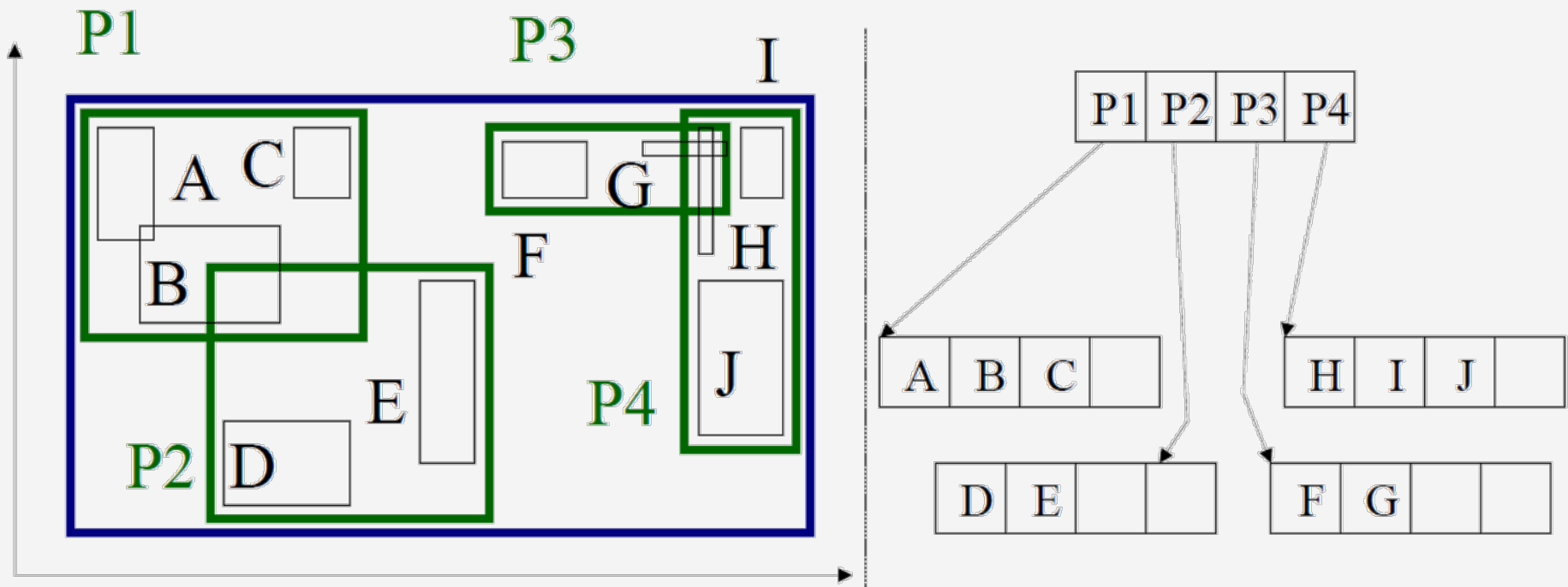


R-δένδρα

- π.χ. με $F = 4$

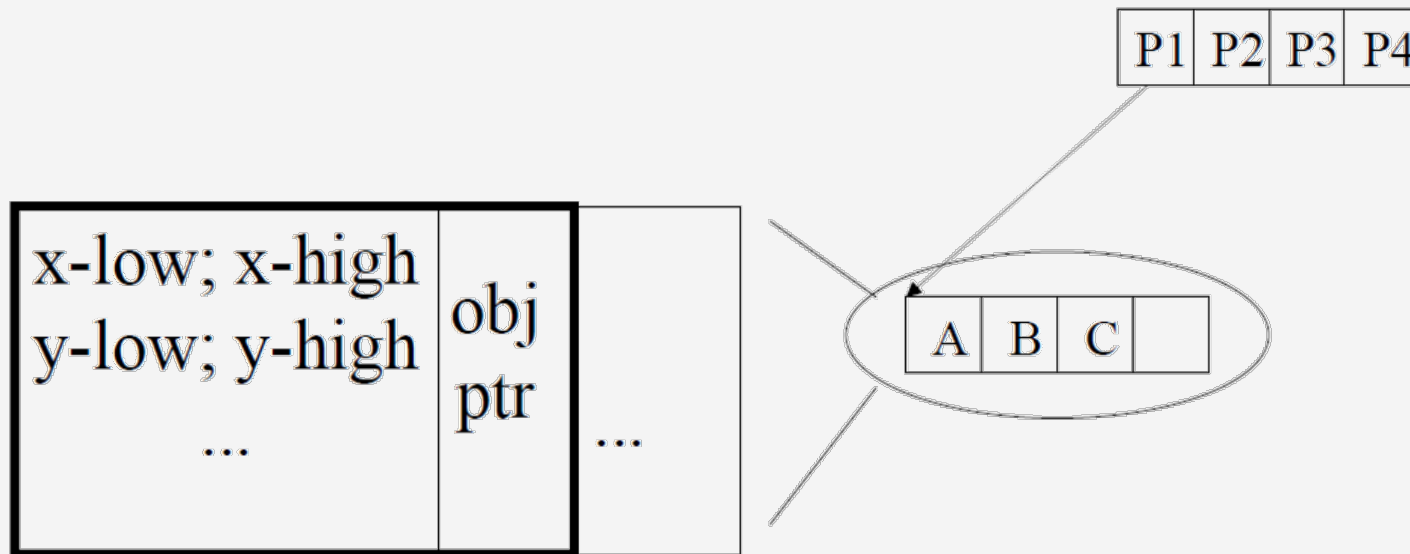


R-δένδρα



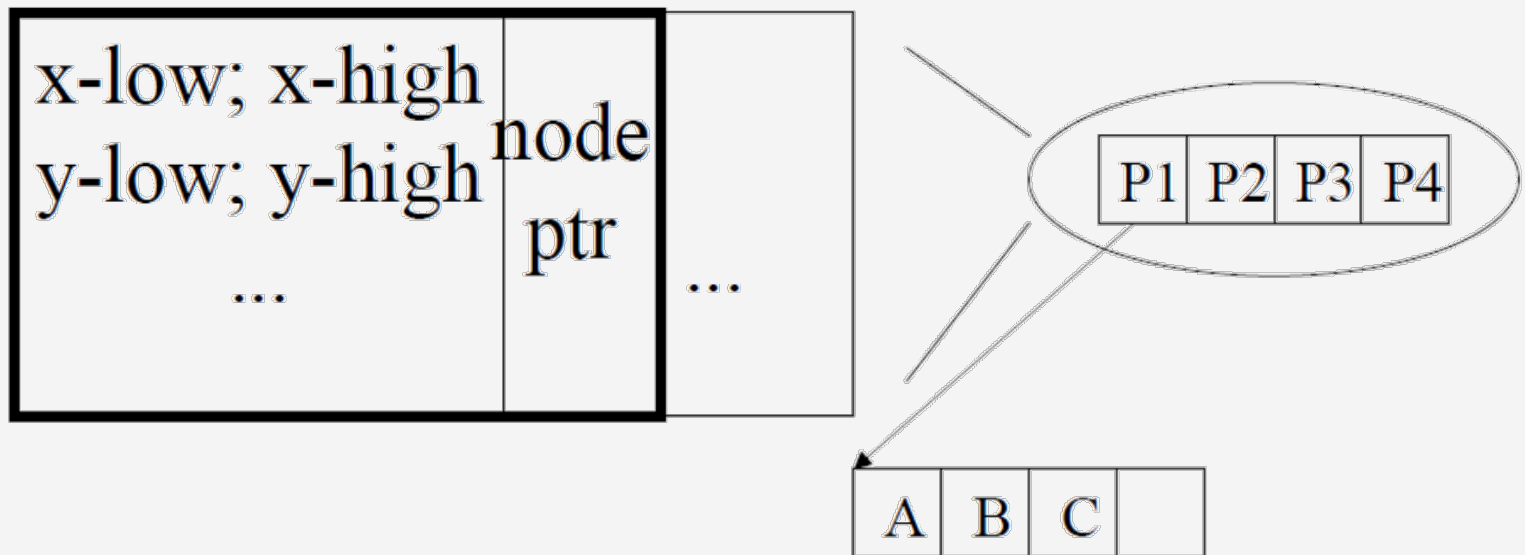
R-δένδρα: Οργάνωση κόμβων

- {(MBR; obj-ptr)} για φύλλα

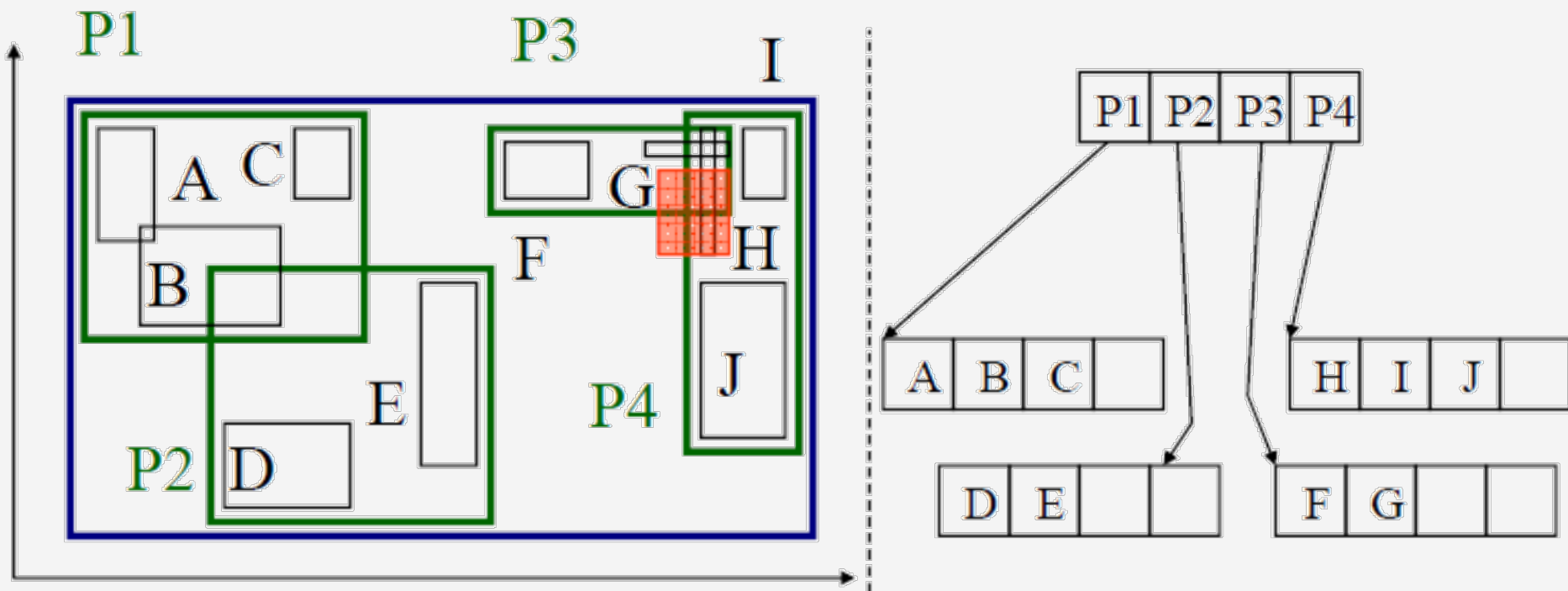


R-δένδρα: Οργάνωση κόμβων

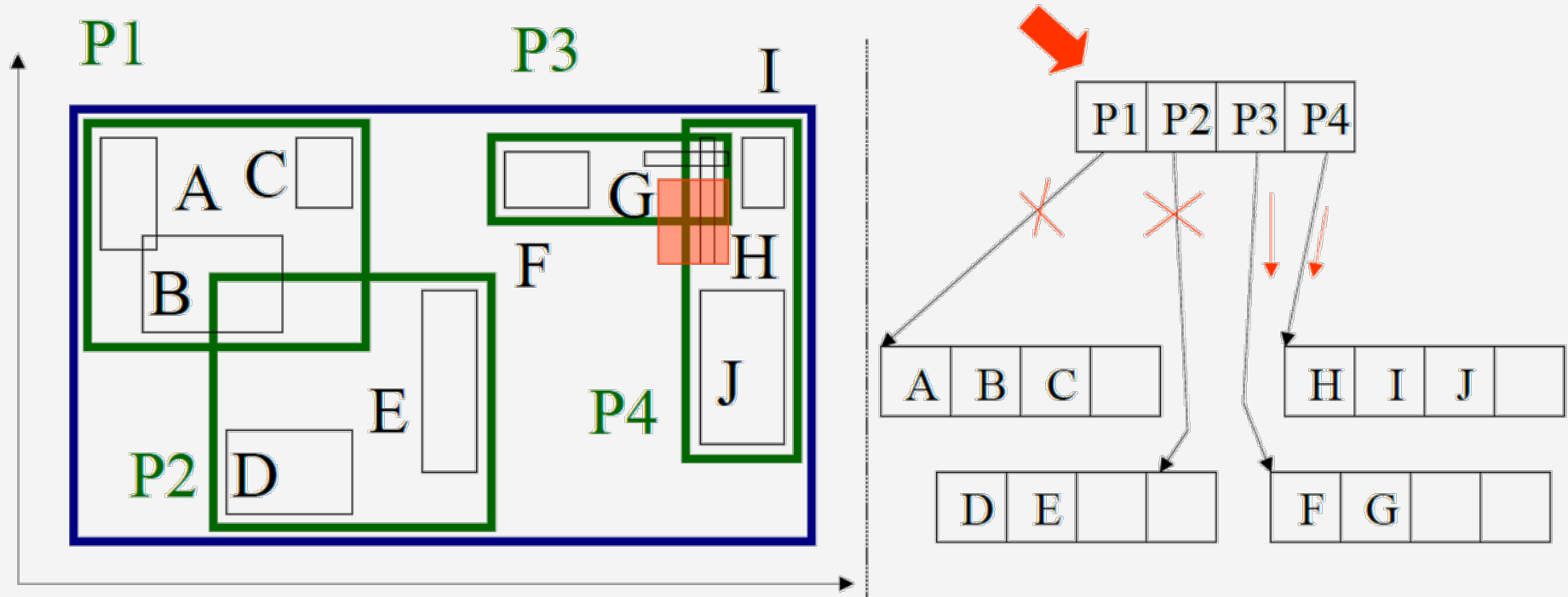
- {(MBR; node-ptr)} για μη-φύλλα



R-δένδρα: αναζήτηση εύρους



R-δένδρα: αναζήτηση εύρους

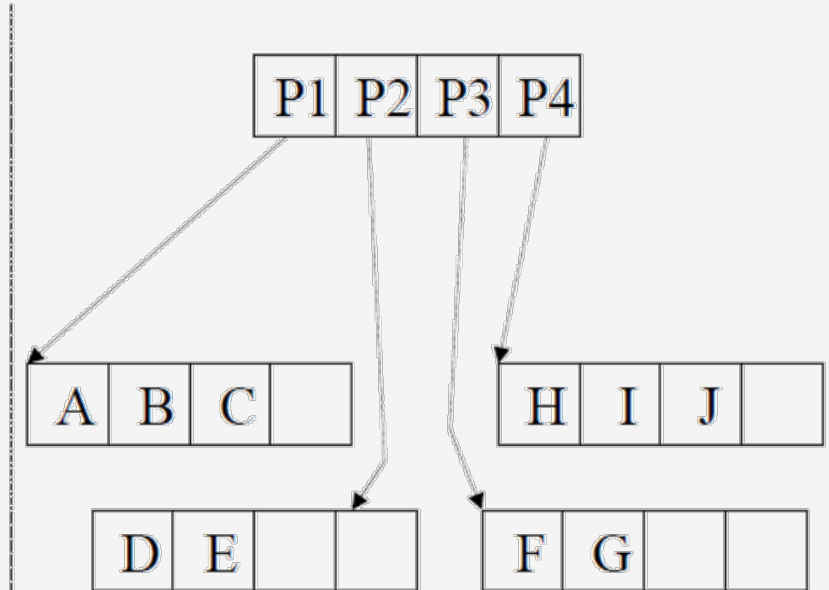
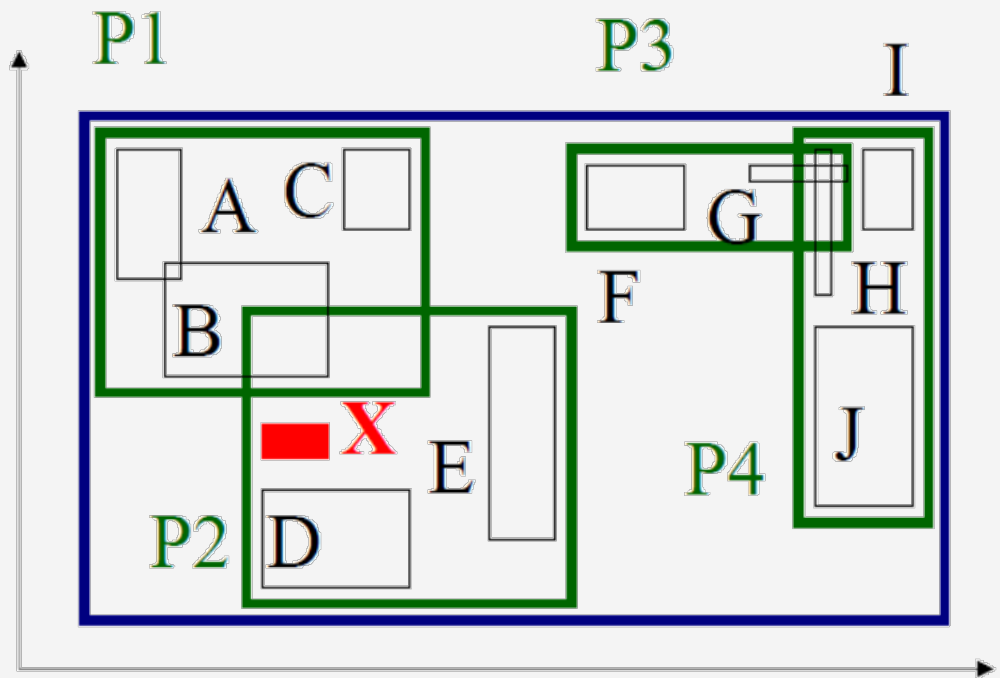


R-δένδρα: αναζήτηση εύρους

- Παρατηρήσεις:
 - Κάθε κόμβος γονέας καλύπτει πλήρως τα παιδιά του
 - Ένα MBR παιδί μπορεί να καλύπτεται από περισσότερους από έναν πατέρα–αποθηκεύεται κάτω από ΜΟΝΟ ΈΝΑΝ από αυτούς. (π.χ., δεν χρειάζεται εξάλειψη διπλοτύπων)
 - Ένα σημειακό ερώτημα μπορεί να ακολουθεί πολλές διακλαδώσεις.
 - Όλα δουλεύουν για **κάθε** διάσταση

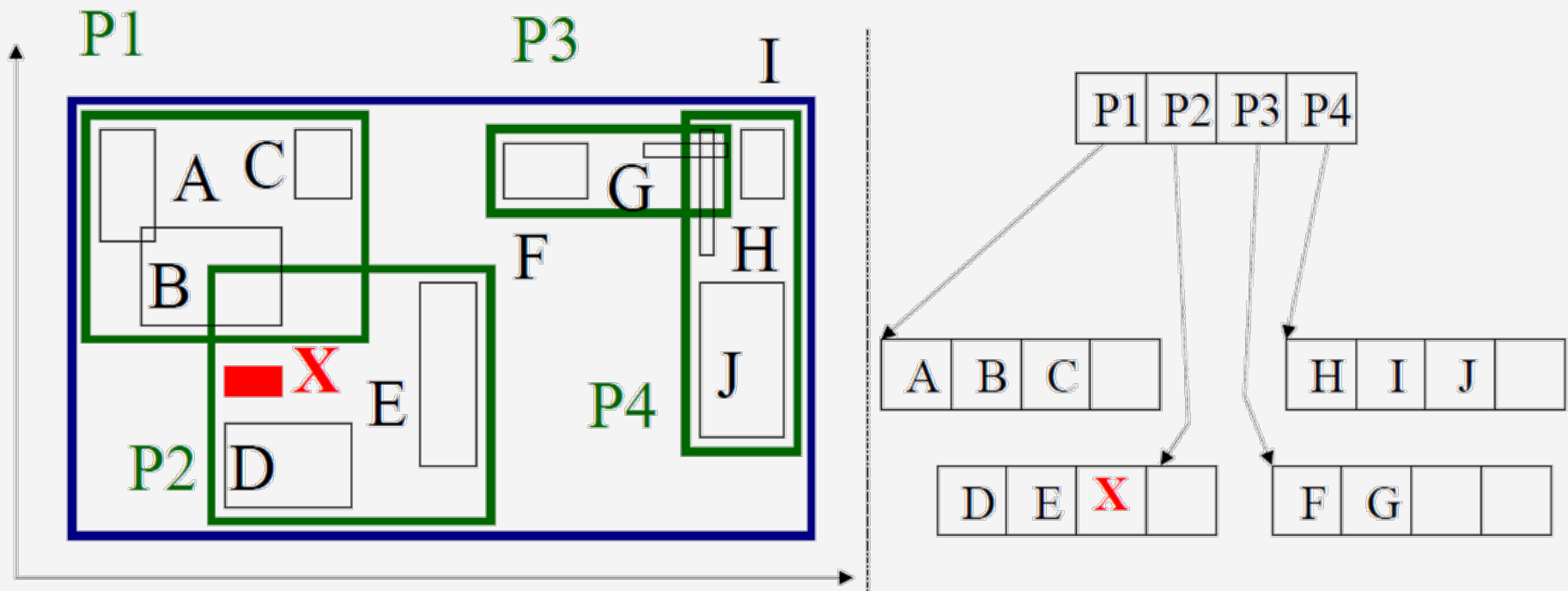
R-δένδρα: εισαγωγή

- π.χ. ορθογώνιο 'X'



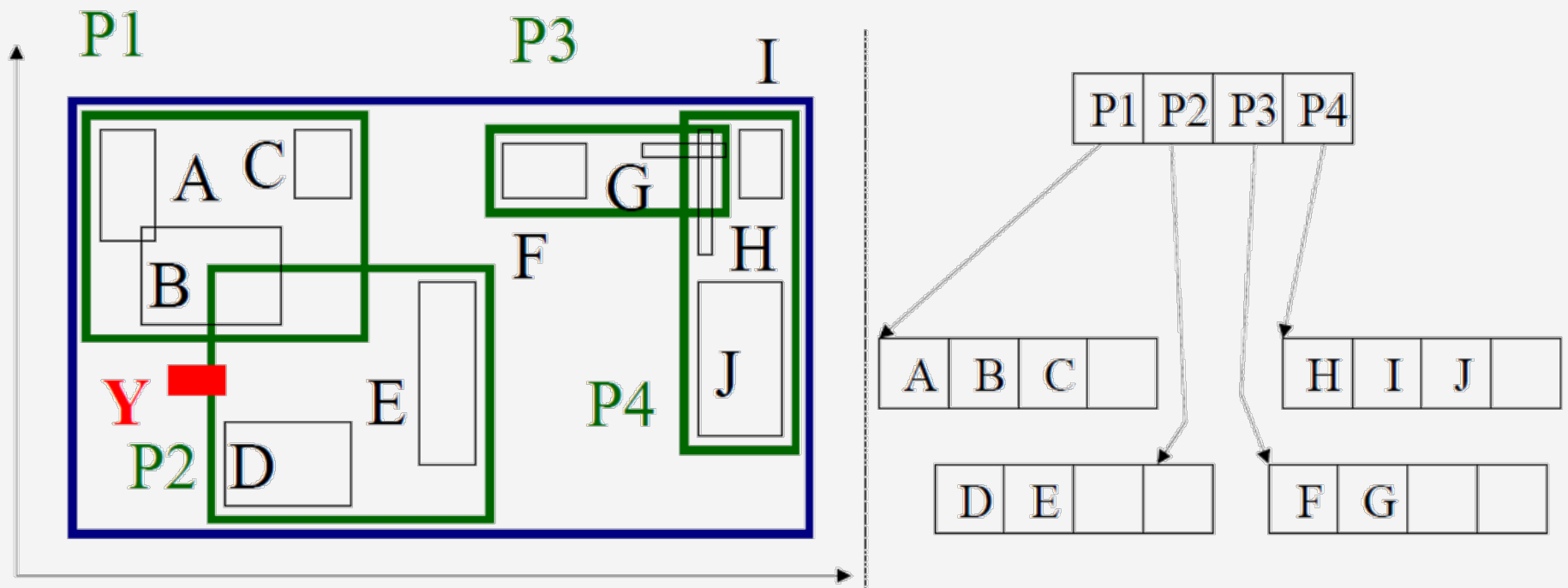
R-δένδρα: εισαγωγή

- π.χ. ορθογώνιο 'X'



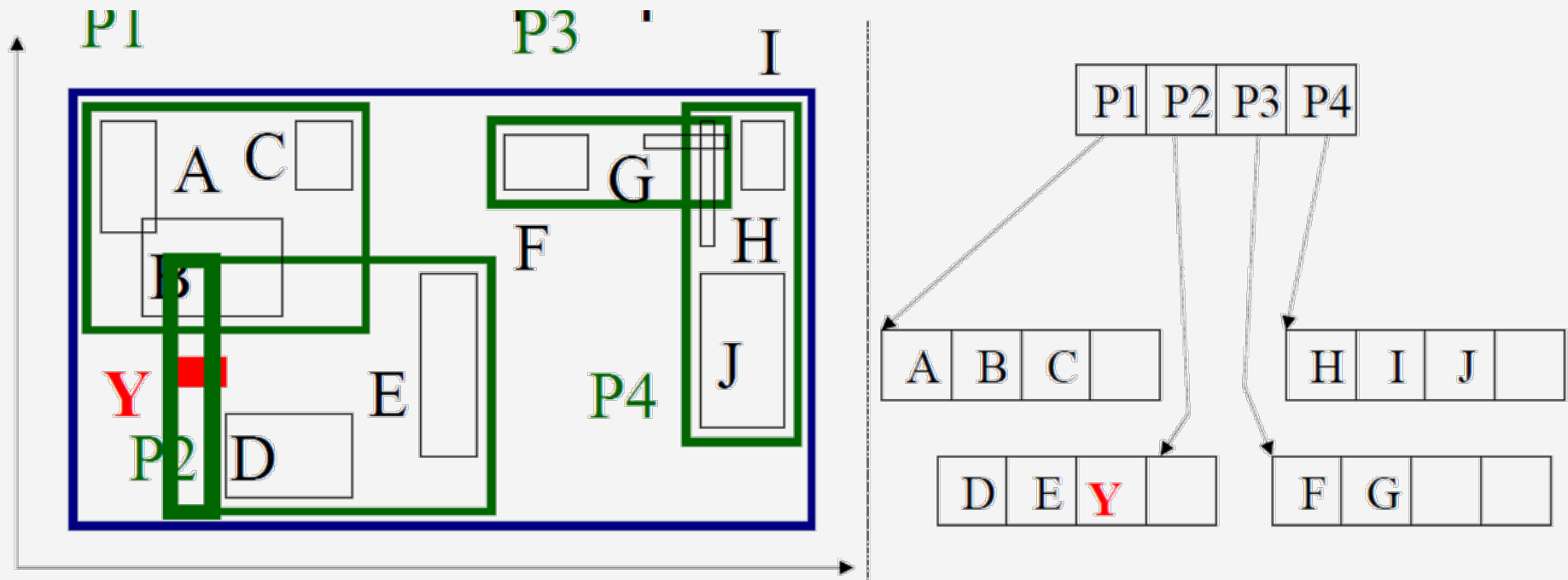
R-δένδρα: εισαγωγή

- π.χ. ορθογώνιο 'Υ'



R-δένδρα: εισαγωγή

- π.χ. ορθογώνιο 'Y': Επέκτεινε τον κατάλληλο γονέα

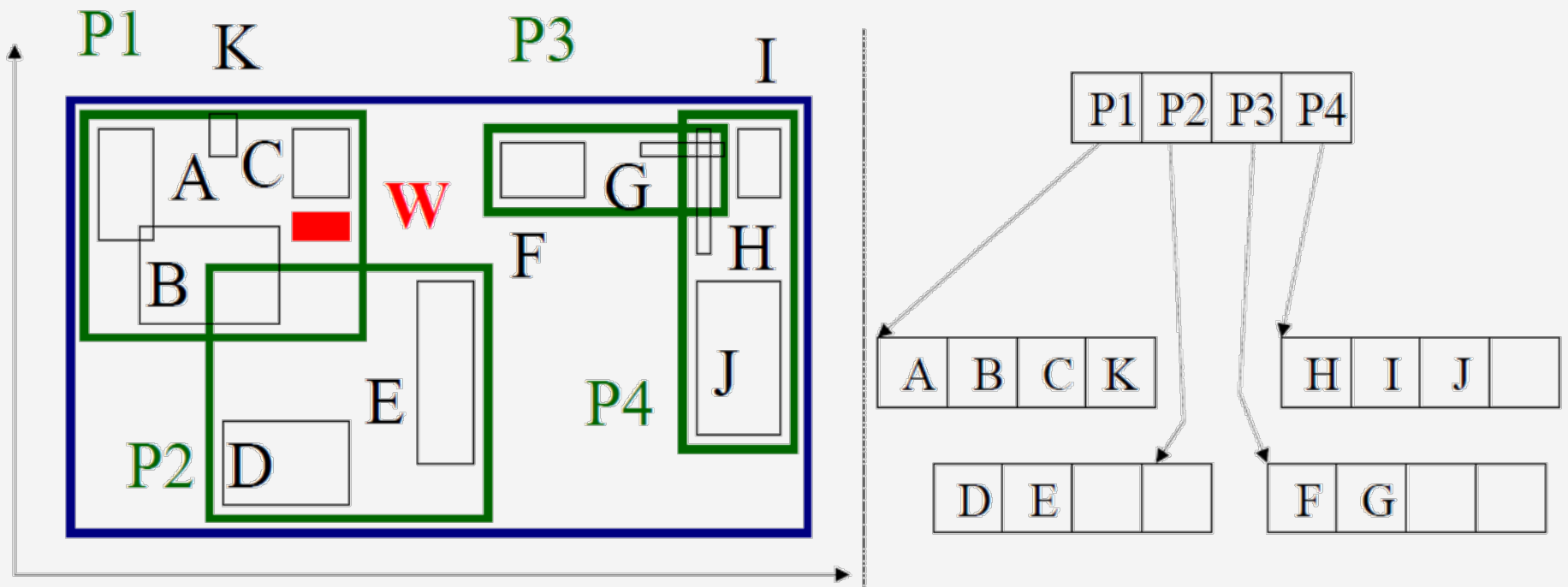


R-δένδρα: εισαγωγή

- π.χ. ορθογώνιο 'Υ': Επέκτεινε τον κατάλληλο γονέα
- Πώς μετράμε την 'καταλληλότητα';
 - Βρες το κατάλληλο που χρειάζεται την ελάχιστη επέκταση για να συμπεριλάβει το Υ.
 - Σε περίπτωση ισοπαλίας, επιλέγουμε το μικρότερο

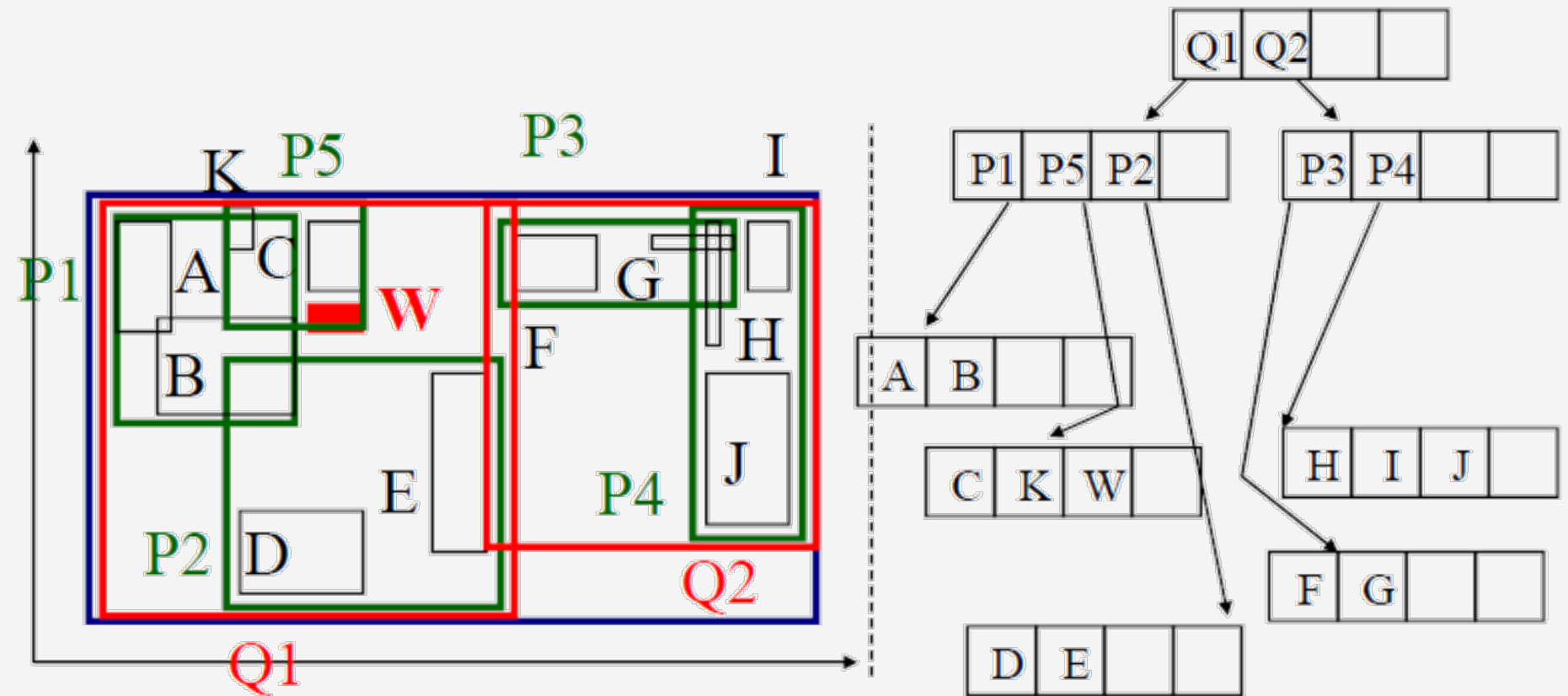
R-δένδρα: εισαγωγή

- π.χ. ορθογώνιο 'W'



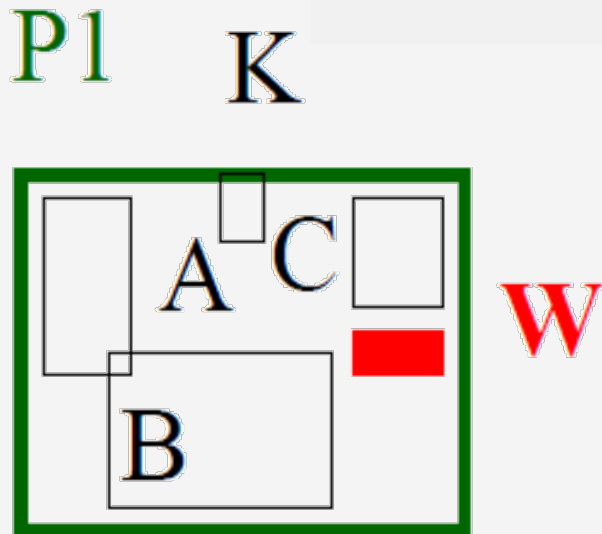
R-δένδρα: εισαγωγή

- π.χ. ορθογώνιο 'W'



R-δένδρα: εισαγωγή

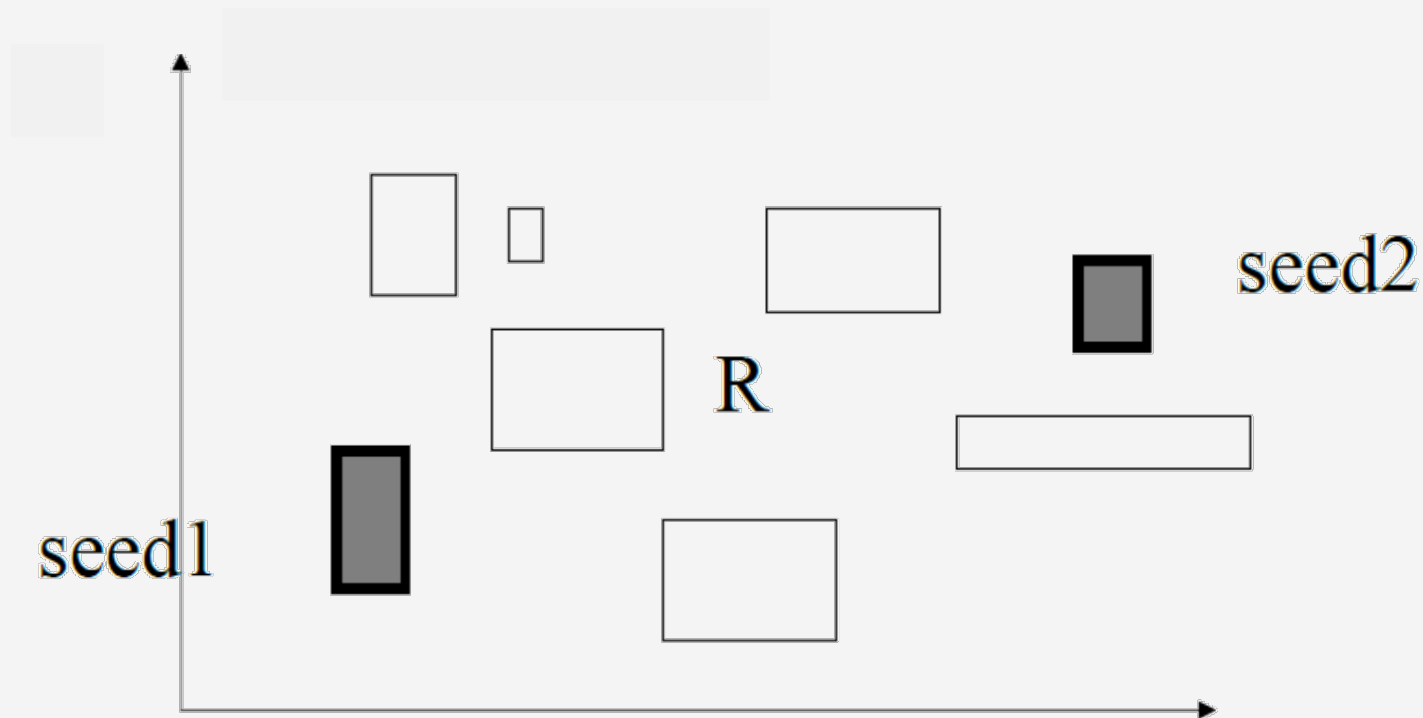
- π.χ. ορθογώνιο 'W': Επικεντρώστε στο P1. Πώς να διαχωριστεί;



- A1: plane sweep, Μέχρι το 50% του ορθογωνίου)
- A2: 'linear' split
- A3: quadratic split
- A4: exponential split

R-δένδρα: εισαγωγή & split

- Επιλέξτε δύο ορθογώνια ως ‘πηγές’
- Αναθέστε κάθε ορθογώνιο ‘R’ στη ‘πλησιέστερη’ ‘πηγή’

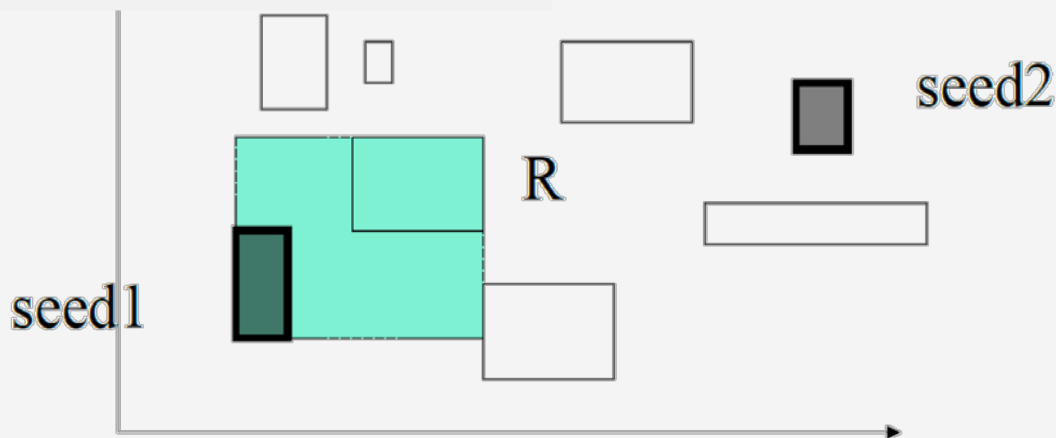


R-δένδρα: εισαγωγή & split

- Επιλέξτε δύο ορθογώνια ως ‘πηγές’
- Αναθέστε κάθε ορθογώνιο ‘R’ στη ‘πλησιέστερη’ ‘πηγή’
- Πώς μετράμε την ‘εγγύτητα’?
- Αυξάνοντας τη περιοχή (ένταση)

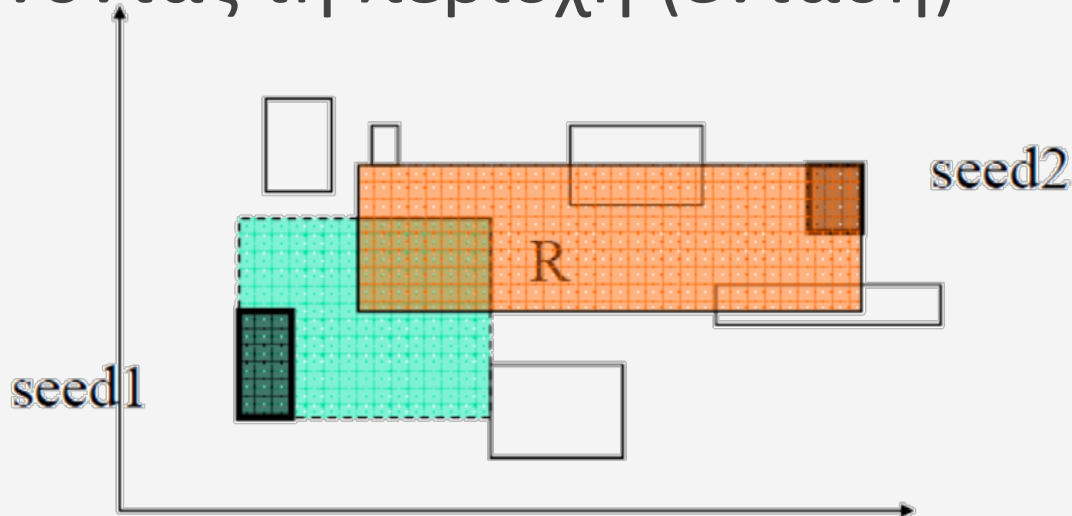
R-δένδρα: εισαγωγή & split

- Επιλέξτε δύο ορθογώνια ως ‘πηγές’
- Αναθέστε κάθε ορθογώνιο ‘R’ στη ‘πλησιέστερη’ ‘πηγή’
- Πώς μετράμε την ‘εγγύτητα’?
- Αυξάνοντας τη περιοχή (ένταση)



R-δένδρα: εισαγωγή & split

- Επιλέξτε δύο ορθογώνια ως ‘πηγές’
- Αναθέστε κάθε ορθογώνιο ‘R’ στη ‘πλησιέστερη’ ‘πηγή’
- Πώς μετράμε την ‘εγγύτητα’?
- Αυξάνοντας τη περιοχή (ένταση)



Πώς επιλέγουμε 'πηγές';

- Quadratic: Για κάθε ζεύγος $E1$ και $E2$, υπολόγισε το ορθογώνιο $J = MBR(E1, E2)$ και $d = J - E1 - E2$. Επέλεξε το ζευγάρι με το μεγαλύτερο d .

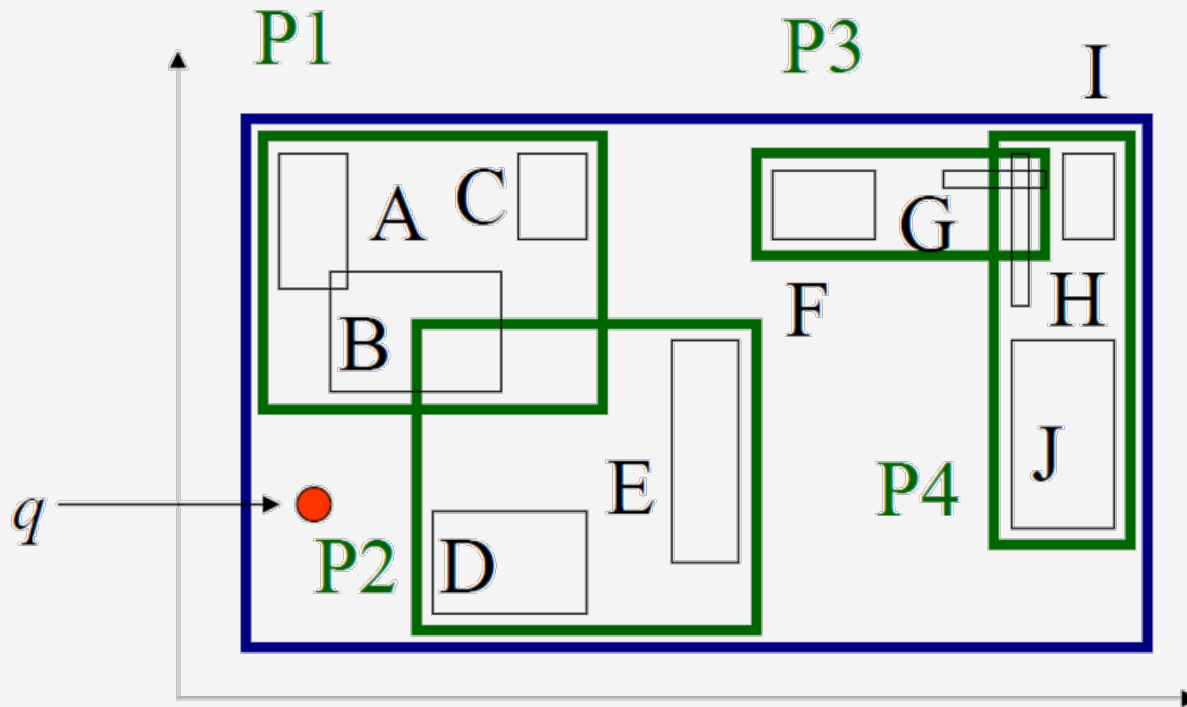
R-δένδρα: εισαγωγή & split

- Επιλέξτε δύο ορθογώνια ως ‘πηγές’
- Αναθέστε κάθε ορθογώνιο ‘R’ στη ‘πλησιέστερη’ ‘πηγή’
- Επιλέξτε σε ποιον πατέρα θα εισάγετε το νέο ορθογώνιο (‘πλησιέστερο’ πατέρα)
- Αν υπερχειλίζει, διαχωρίστε σε δύο, χρησιμοποιώντας (έστω) τον quadratic split αλγόριθμο
 - Διαδώστε το διαχωρισμό προς τα πάνω,
- αν χρειάζεται ενημερώστε τους MBRs των επηρεαζόμενων πατέρων.

R-δένδρα: Διαγραφή

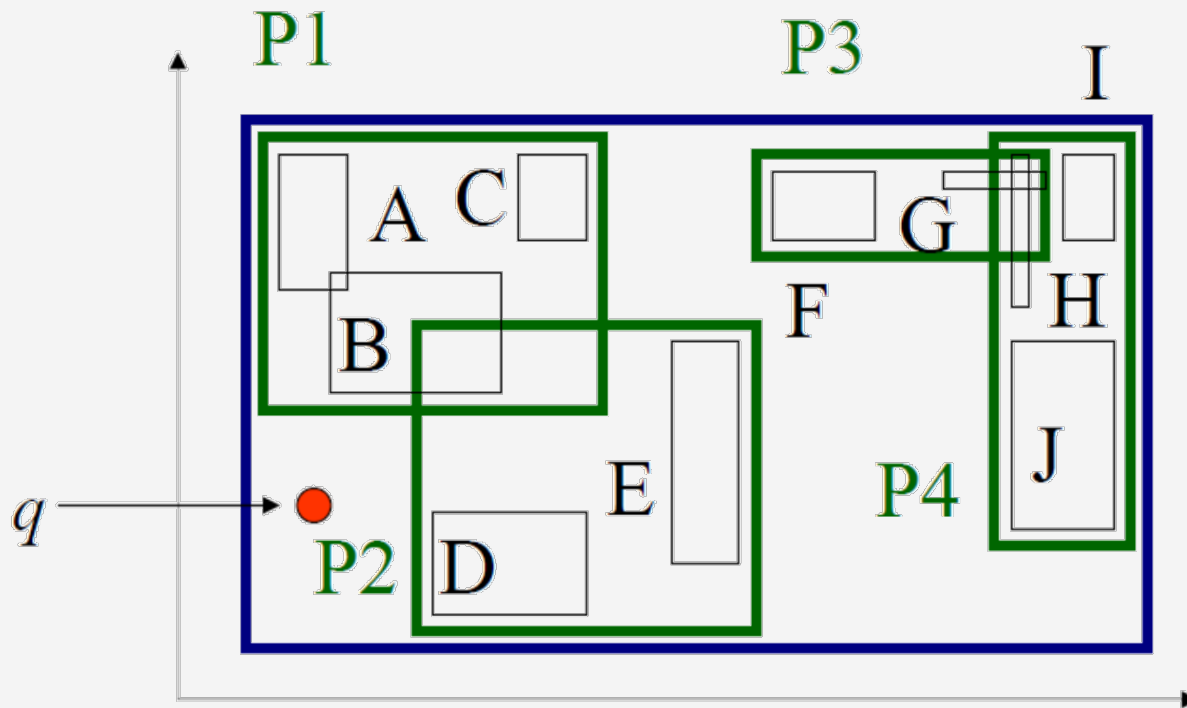
- Διαγραφή ορθογωνίου
- Αν υπερχειλίζει
 - Προσωρινά διαγράψτε όλα τους συγγενείς(!)
 - Διαγράψτε τον πατρικό κόμβο και
 - Επανα-εισάγετε τους

R-δένδρα: NN αναζήτηση



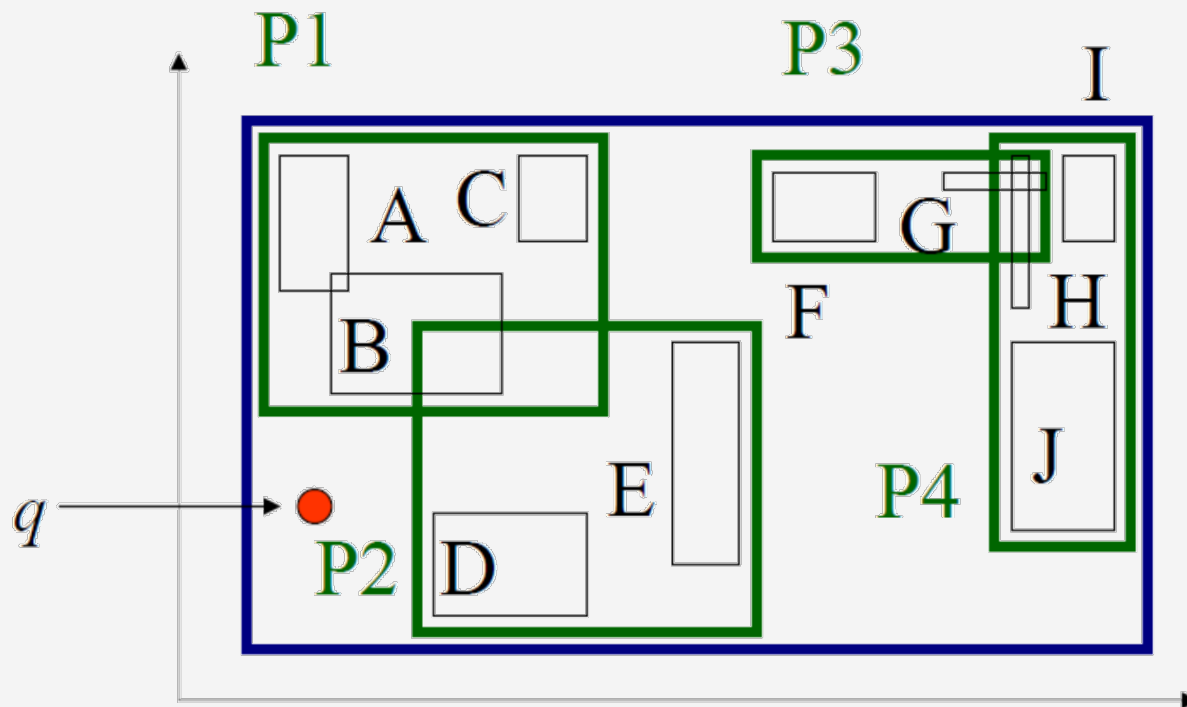
R-δένδρα: NN αναζήτηση

- Πώς;
 - Βρες τον πλησιέστερο γείτονα, εκλέπτυνε...



R-δένδρα: NN αναζήτηση

- Μία προσέγγιση:
 - Ουρές προτεραιότητας, με υποσχόμενα MBRs και την βέλτιστη και χείριστη απόσταση
 - Θεωρούμε μόνο τα P2 και P4, χάριν παραδείγματος

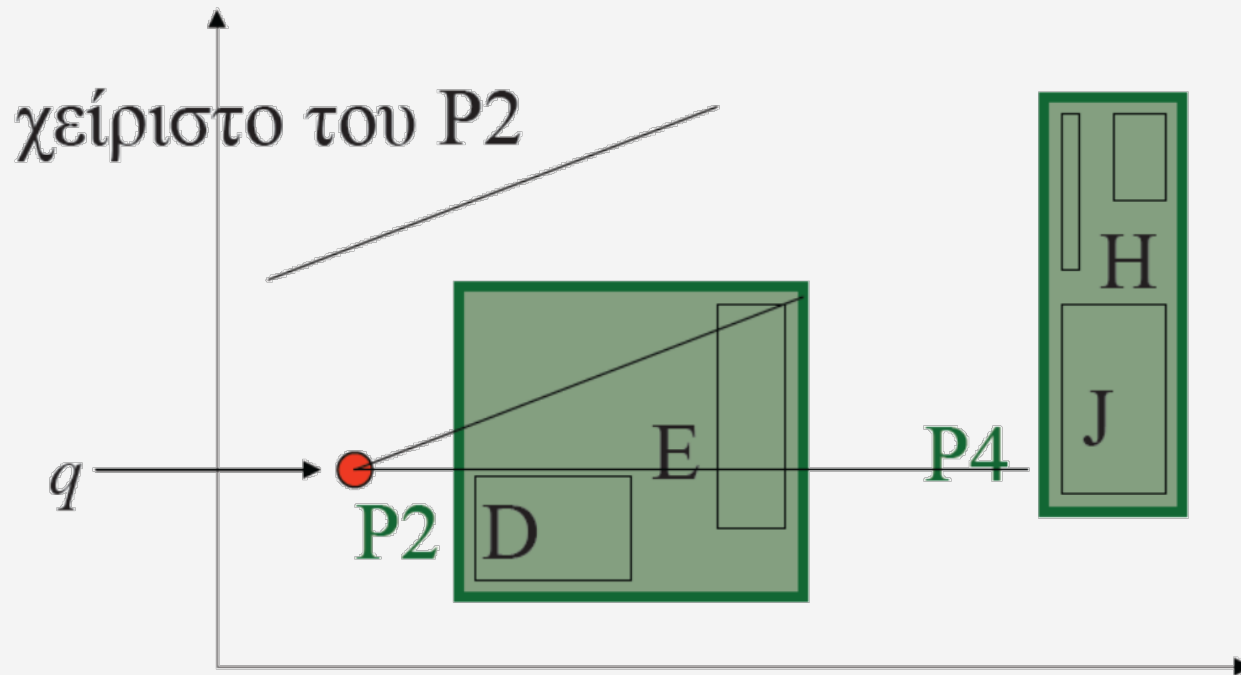


R-δένδρα: NN αναζήτηση

βέλτιστο του P4

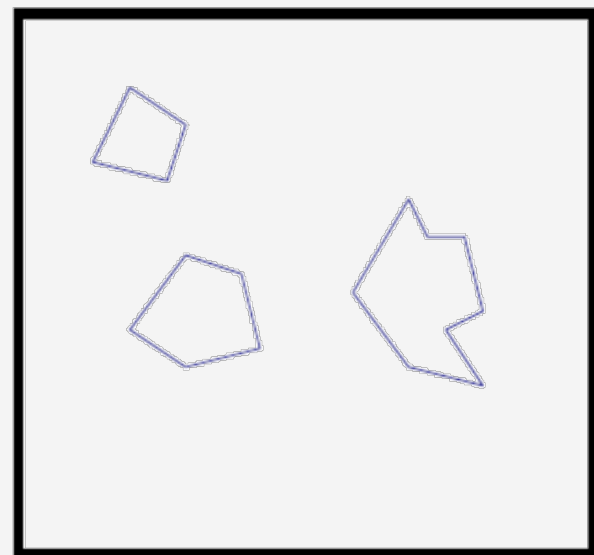
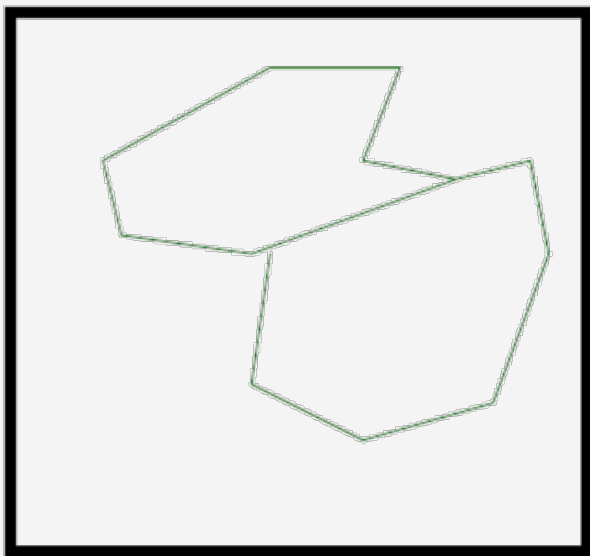
=> P4 είναι άχρηστο

για 1-nn



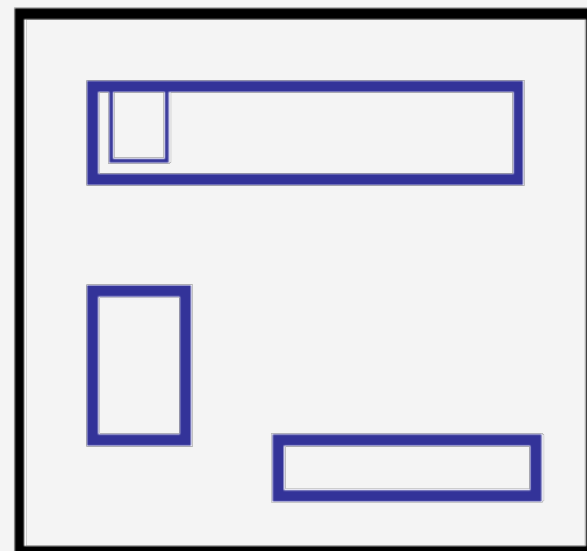
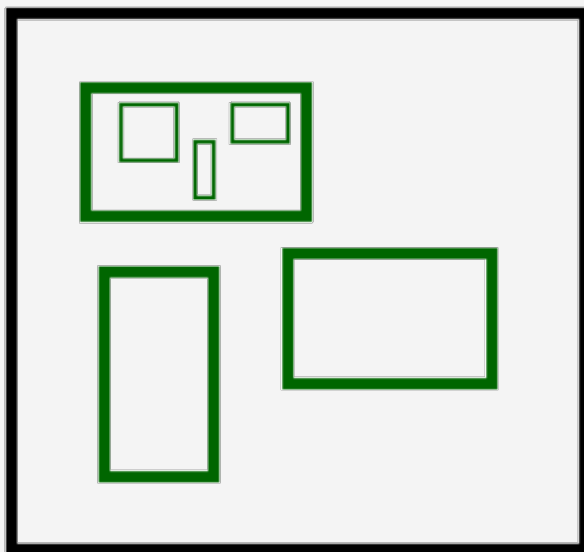
R-δένδρα: Spatial joins

- Βρες όλους του νομούς που τέμνουν τις λίμνες



R-δένδρα: Spatial joins

- Υπέθεσε ότι είναι οργανωμένα σε R-δέντρα



R-δένδρα: Spatial joins

- Για κάθε πατέρα P1 του δένδρου T1
 - Για κάθε πατέρα P2 του δένδρου T2
 - Αν το MBRs τους τέμνει,
 - Επεξεργαστείτε τα αναδρομικά (πχ., ελέγξτε τα παιδιά τους)
- Βελτιώσεις-παραλλαγές:
 - [Seeger+, sigmod92]: κάνε κάποια προ-επεξεργασία, κάνε plane-sweeping για να αποφύγετε $N1 * N2$ ελέγχου για τομές
 - [Lo & Ravishankar, sigmod94]: 'seeded' R-trees (πολλές δημοσιεύσεις πάνω σε spatial joins, χωρίς R-trees: [Koudas+ Sevcik], κτλ.)

Παραλλαγές των R-δένδρων

- R+-tree: Δεν επιτρέπεται η επικάλυψη μεταξύ φύλλων και κόμβων, και υιοθετείται η τεχνική της αποκοπής (clipping)
- R*-tree: Βασίζεται στο αρχικό R-tree με μία παραλλαγή στον αλγόριθμο εισαγωγής του.
- Packed Hilbert R-tree: Κατασκευή bottom – up, έχοντας πρώτα διατάξει τα αντικείμενα που θα εισαχθούν κατά τη σειρά Hilbert
- Αλγόριθμοι:
 - Αναζήτηση σημείου, εύρους, πλησιέστερου γείτονα
 - Χωρική σύνδεση
 - Τοπολογικές αναζητήσεις
- Εφαρμογές:
 - Χωρικές Βάσεις
 - Χωροχρονικές Βάσεις - Κινούμενα αντικείμενα
 - Βάσεις πολυμέσων

Σύνοψη

- Τύποι διατεταγμένων ευρετηρίων ενός επιπέδου
 - Πρωτεύοντα ευρετήρια
 - Ευρετήρια Συστάδες
 - Δευτερεύοντα Ευρετήρια
- Ευρετήρια πολλών επιπέδων
- Δυναμικά ευρετήρια πολλών επιπέδων με χρήση B-δένδρων και B⁺-δένδρων
- Ευρετήρια σε πολλά κλειδιά
- Ευρετήρια χωρικών δεδομένων
 - R-δένδρα

Αναφορές

- Christos Faloutsos and Ibrahim Kamel. “Beyond Uniformity and Independence: Analysis of R-trees Using the Concept of Fractal Dimension”. Proc. ACM PODS, 1994.
- Yannis Theodoridis and Timos Sellis. “A Model for the Prediction of Rtree Performance”. Proc. ACM PODS, 1996.