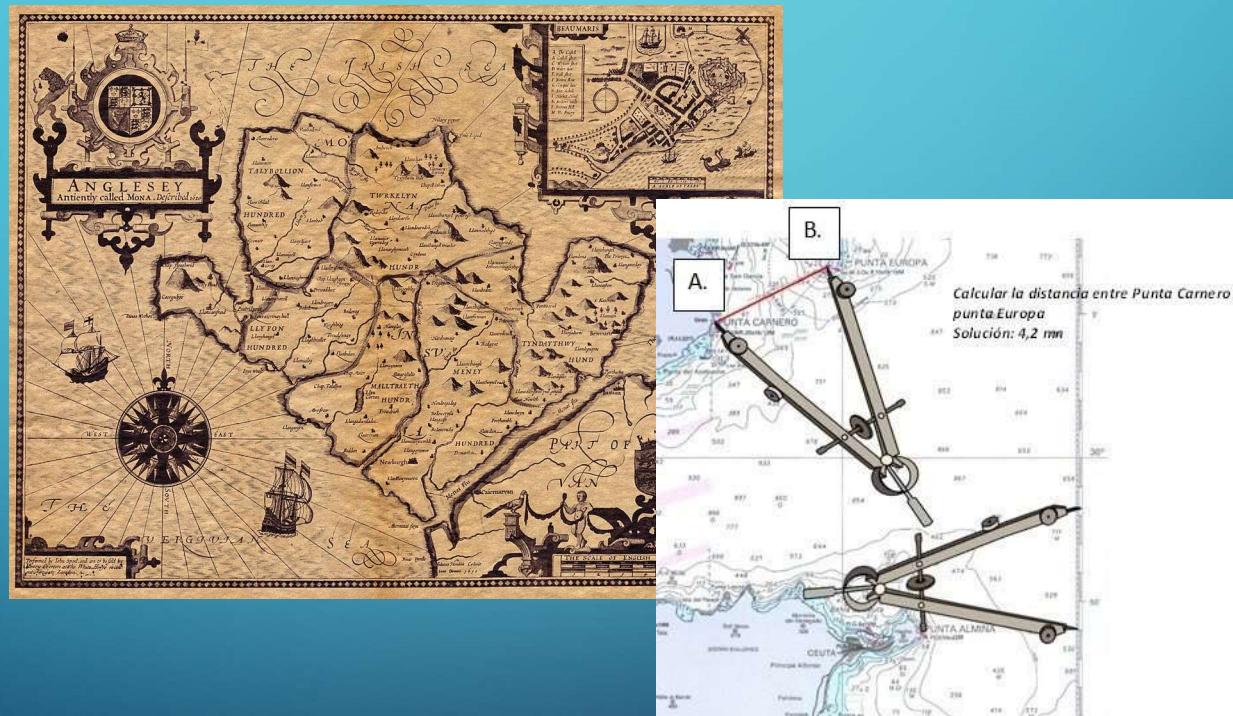


Modulo “Analisis de Datos Cientificos y Geograficos”

DATOS ESPACIALES

LOS ORIGENES...





MAS Y MAS INFORMACION.... EN PAPEL !



LA DOCUMENTACION SE DIGITALIZA...



SE PODRA USAR SQL EN MAPAS ?





SISTEMAS ESPACIALES GEOGRAFICOS (GIS)

Mapean figuras o cuerpos para armar una base de datos

Un objeto geográfico es una entidad representada por

- *una componente espacial: su geometría*
- *una descripción (que depende del tema a tratar)*

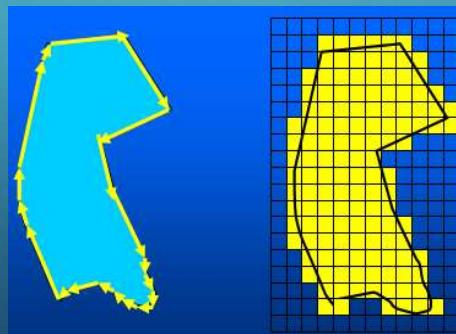
SISTEMAS ESPACIALES GEOGRAFICOS (GIS)

Mapean figuras o cuerpos para armar una base de datos

Un objeto geográfico es una entidad representada por

- *una componente espacial: su geometría*
- *una descripción (que depende del tema a tratar)*

Vector (polígono)

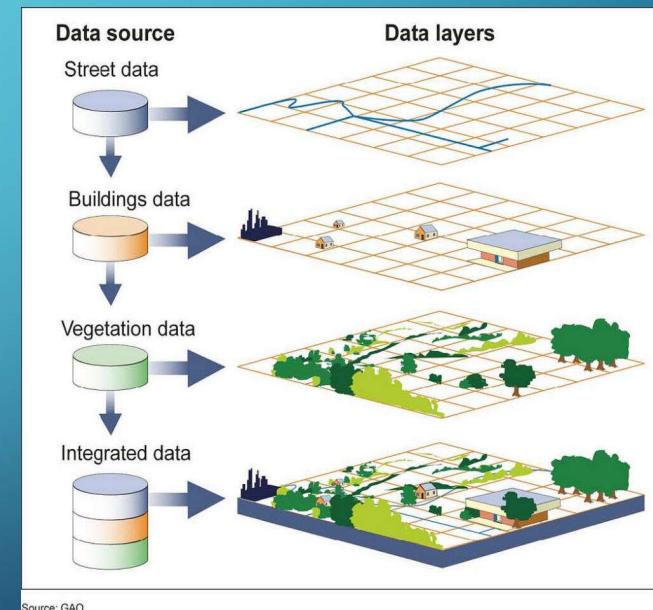


Raster (*mapa de bits/pixels*)
el color representa objetos diferentes o
características distintivas (altura, etc)

SISTEMAS ESPACIALES GEOGRAFICOS (GIS)

De acuerdo a los temas elegidos, un mismo espacio puede contener varios niveles (*layers*), aun compartiendo objetos con la misma geometría.

Ejemplo: un layer representa los ríos de una región, otro layer representa las ciudades, etc.



EJEMPLO DE LAYERS

Nivel A: densidad demográfica

- 1 → 10 hab/Km²
- 2 → 10 hab/Km²
- 3 → 10 hab/Km²

Nivel B: índice de alfabetismo

- 1 → 20%
- 2 → 48%
- 3 → 64%

Nivel C: médicos por habitante

- 1 → 0.6 m/hab
- 2 → 0.8 m/hab
- 3 → 0.8 m/hab



ATENCION !!!

Sistema Espacial \neq Sistema con Referencias Geograficas

Provincia	Poligono	Poblacion
Buenos Aires		2.000.000
Cordoba		950.000
Misiones		160.000

Provincia	Superficie	Poblacion
Buenos Aires	1000	2.000.000
Cordoba	800	950.000
Formosa	450	160.000
...

Un **SRID** (*Spatial Reference System Identifier*) es un identificador estándar único que hace referencia a un *Sistema de Coordenadas* concreto

El **SRID** define todos los parámetros del *Sistema de Coordenadas* y la **proyección** de los datos.

EJEMPLOS DE CONSULTAS ESPACIALES

Sistema Espacial

Provincia	Polígono	Población
Buenos Aires		2.000.000
Cordoba		950.000
Misiones		160.000



EJEMPLOS DE CONSULTAS ESPACIALES

Sistema Espacial

Provincia	Polígono	Población
Buenos Aires		2.000.000
Cordoba		950.000
Misiones		160.000



EJEMPLOS DE CONSULTAS ESPACIALES

Sistema Espacial

Provincia	Polígono	Población
Buenos Aires		2.000.000
Cordoba		950.000
Misiones		160.000



EJEMPLOS DE CONSULTAS ESPACIALES

Sistema Espacial

Provincia	Polígono	Población
Buenos Aires		2.000.000
Cordoba		950.000
Misiones		160.000





OTROS EJEMPLOS DE CONSULTAS ESPACIALES

- Detectar los tweets que fueron emitidos a menos de 5 cuadras del Hilton de Puerto Madero entre las 15 y 17 hs del día de hoy.
- Encontrar pares de bancos en el barrio de Palermo que se hallen a menos de 2 cuadras entre si
- Encontrar todos los templos a menos de 3 cuadras de un salón de fiestas dentro de la CABA
- Detectar los domicilios que se verían afectados en las cercanías de un río, si éste aumentara su ancho en un 10%



SISTEMAS ESPACIALES

Geometría

Raster o Vectorial

Lenguaje de Consulta

Consultas de ventana

Consulta puntual

Consulta de junta espacial **Costosa**

Busqueda basadas en Métrica
Busqueda basadas en Topología

Procesamiento de Consultas

Filtros y Refinamiento

Índices Espaciales

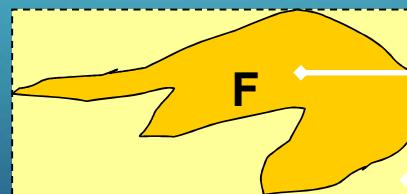
METODOS DE ACCESO ESPACIAL

Indices sobre objetos reales:

Puntual \Rightarrow coordenadas

No puntual \Rightarrow MBB ó MBR

**MBB
ó
MBR**

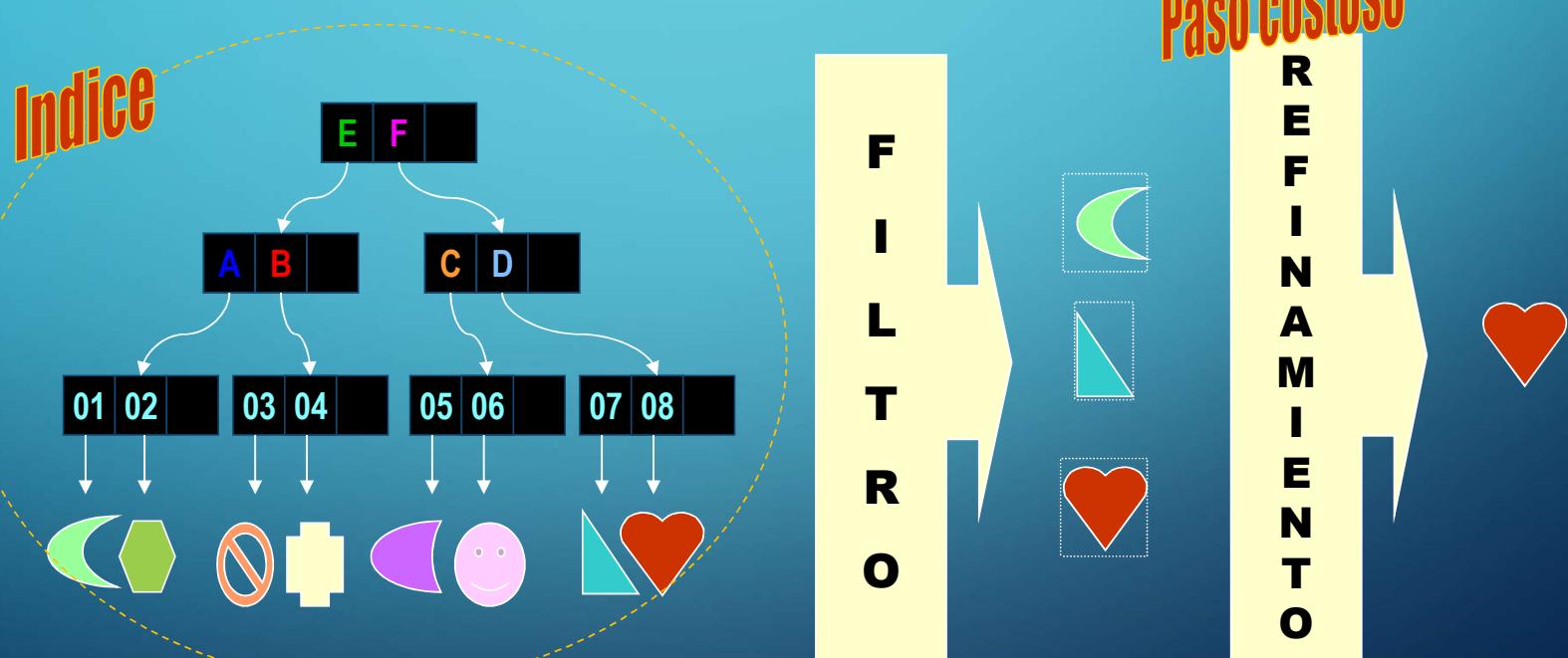


geometría real de F

mbb asociado a F

SISTEMAS ESPACIALES

Una consulta es el resultado de dos pasos:





INDICES PARA SISTEMAS ESPACIALES



Space-Drive Index

vs.

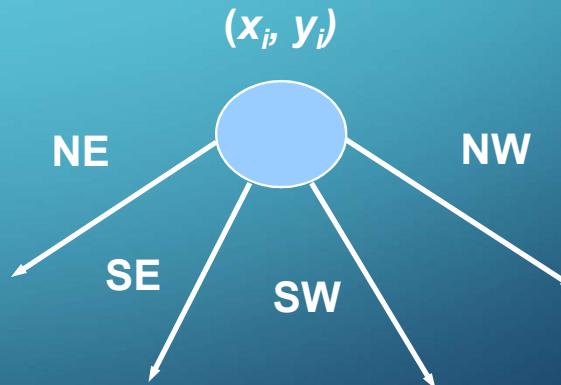
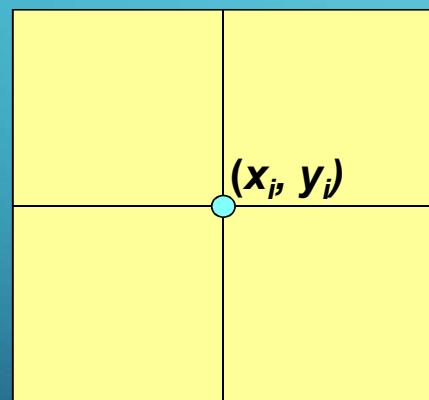
QuadTree

Data-Driven Index

R-Tree

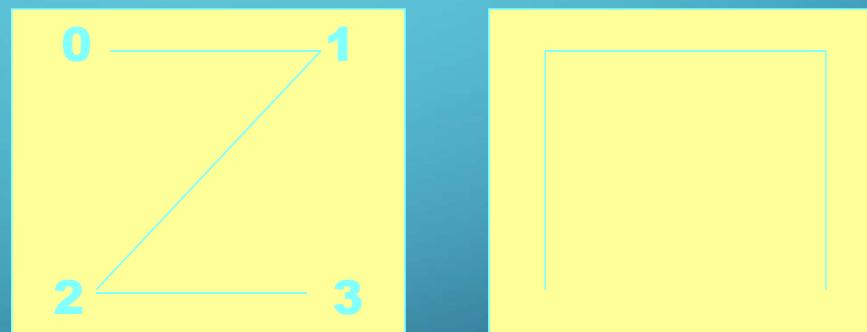
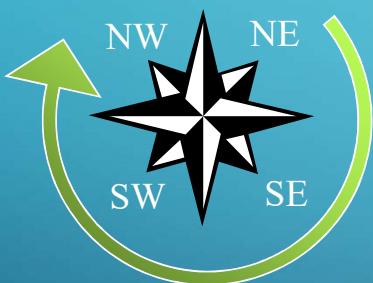
INDICE QUADTREE

Estructura de datos que permite indexar regiones en dos dimensiones con resolución variable (más detalle en las regiones con mayor densidad)



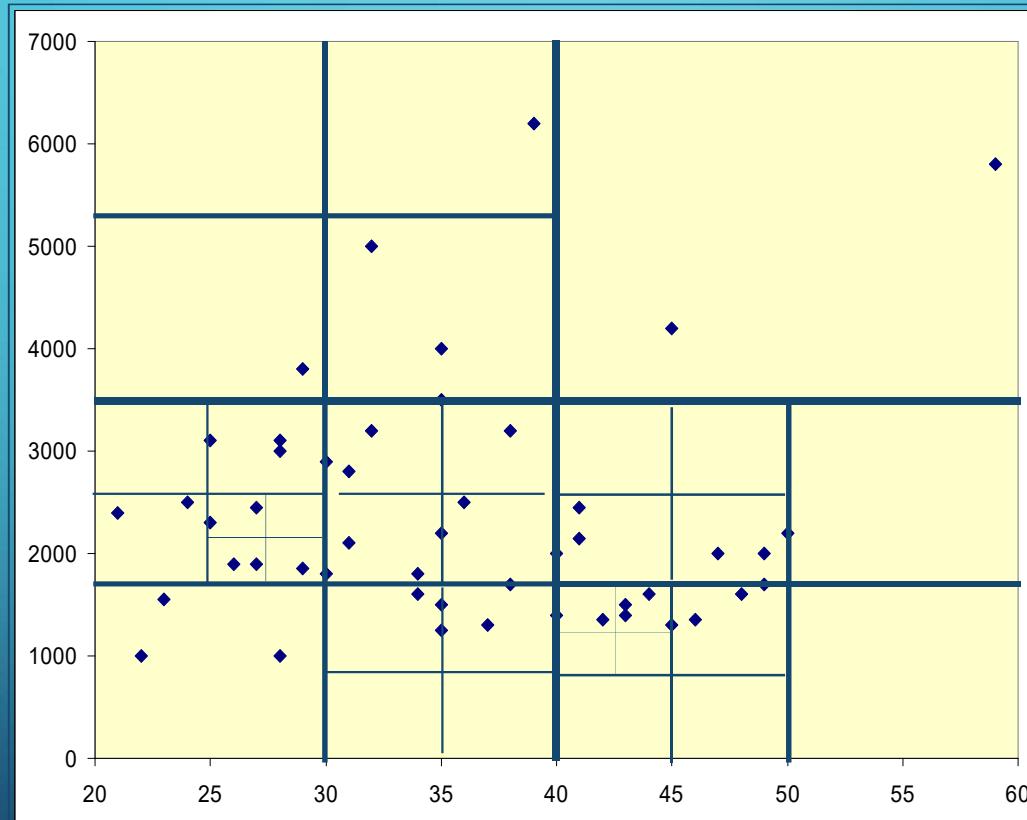
INDICE QUADTREE

- Cada cuadrante es representado por un nodo
- Cada nodo representa un *rango* de valores
- Cada nodo no hoja tiene 4 hijos (subcuadrante)

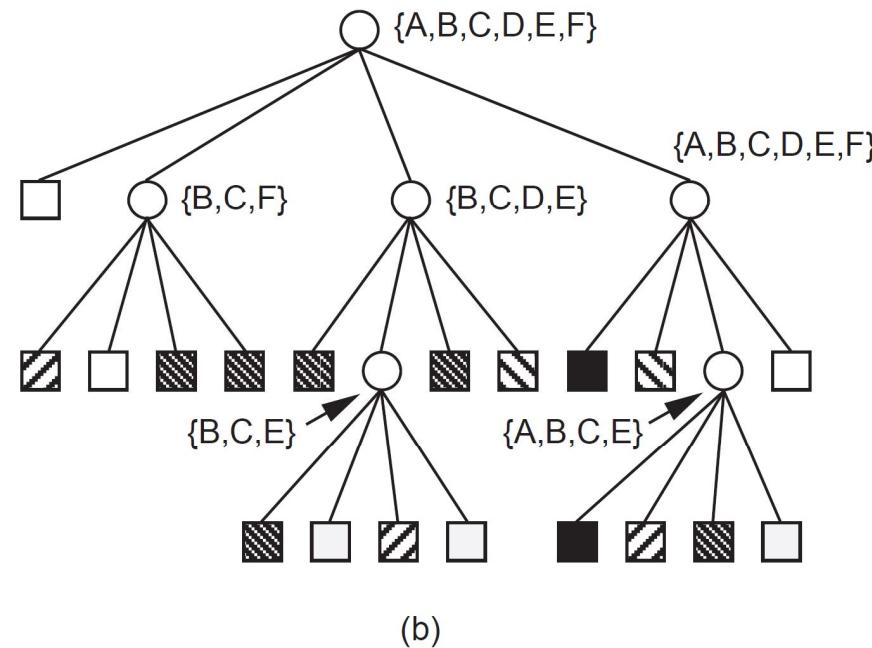
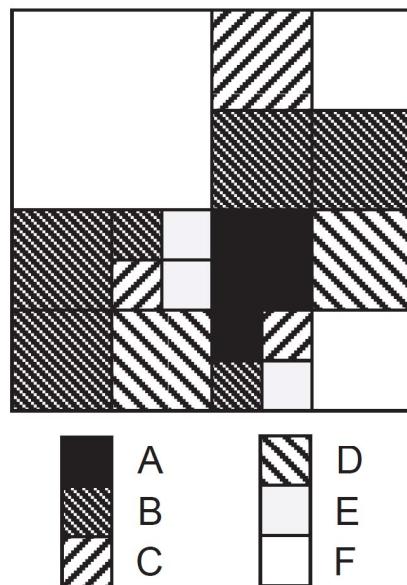


*La división es siempre a la mitad del rango,
para todas las dimensiones*

QUADTREE POR DENSIDAD

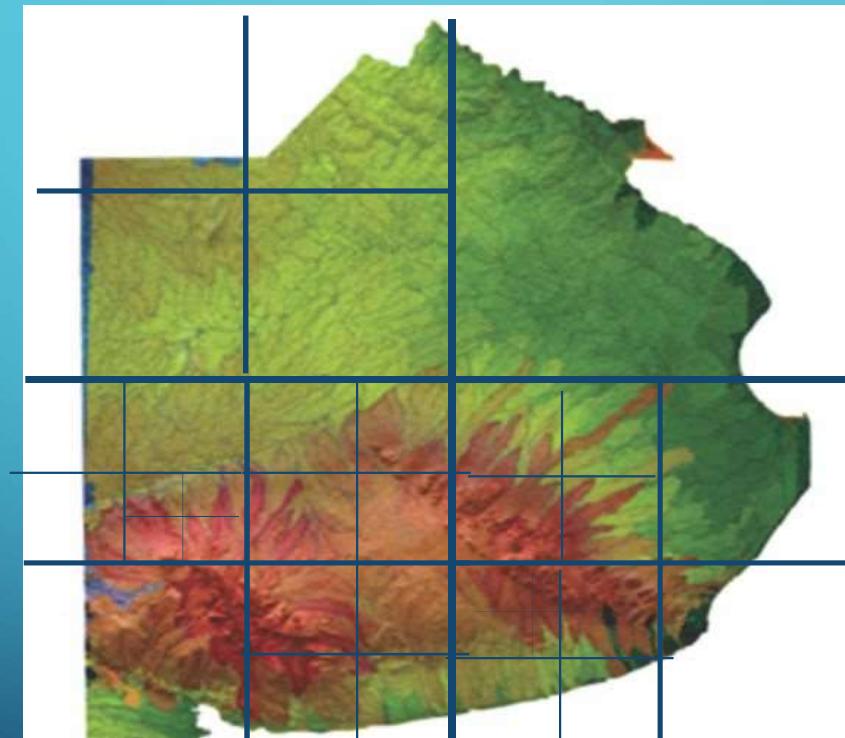


QUADTREE POR HOMOGENEIDAD



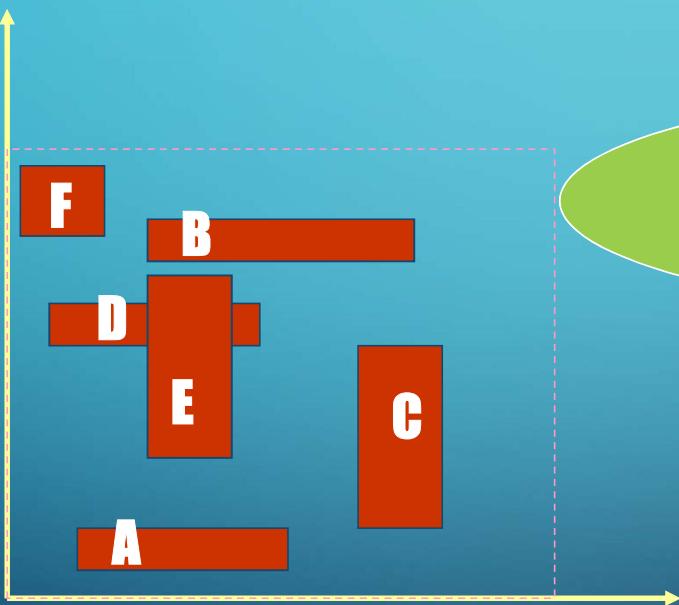


QUADTREE POR HOMOGENEIDAD



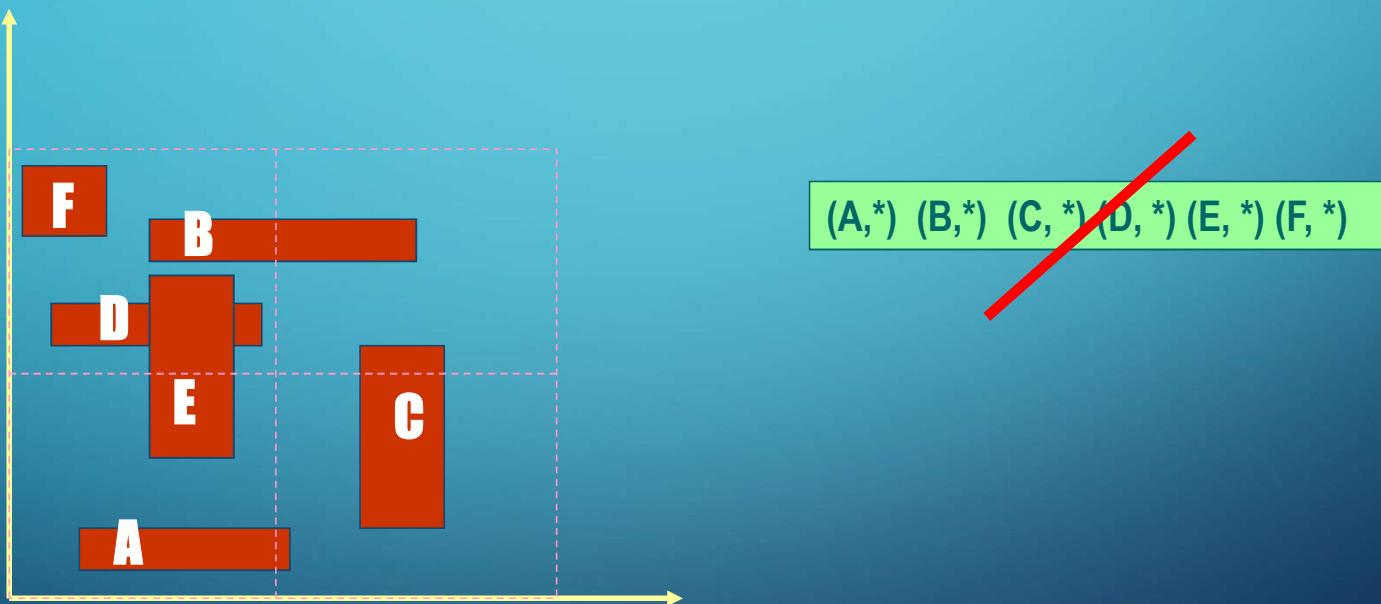
QUADTREE PARA GIS

En las hojas se almacena el RID de los objetos cuyo MBR tiene intersección con su zona espacial

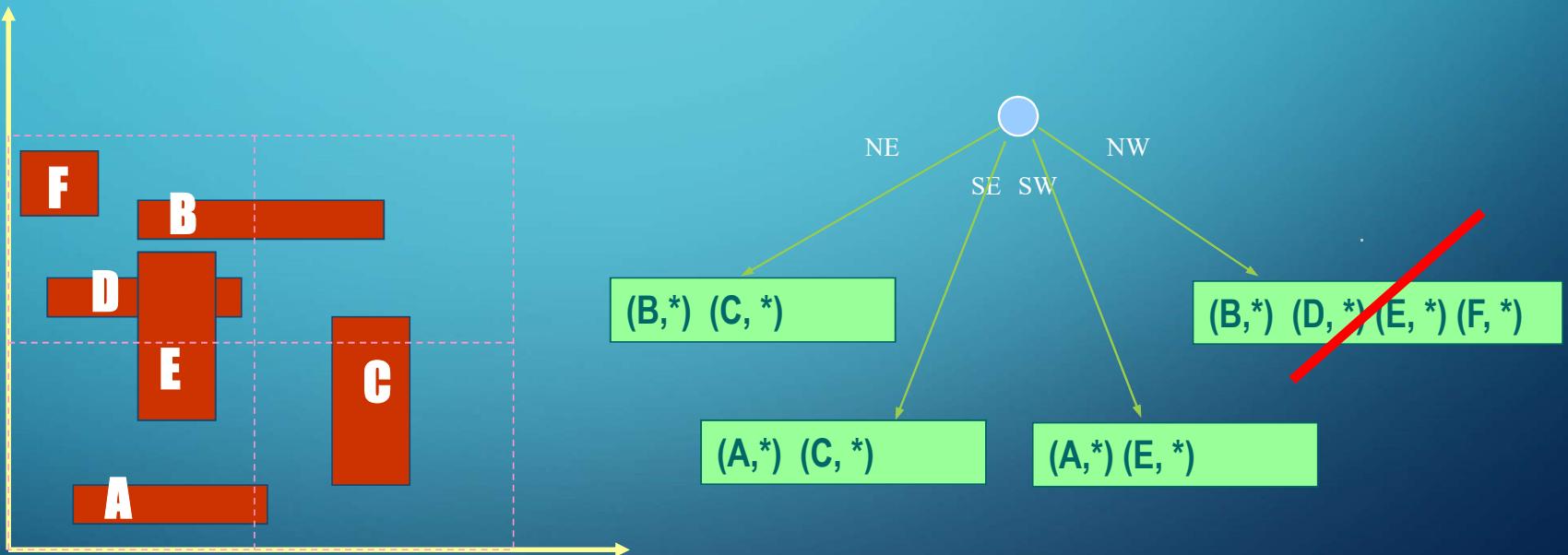


Ejemplo: 2D y bloques de disco
con capacidad para tres tuplas

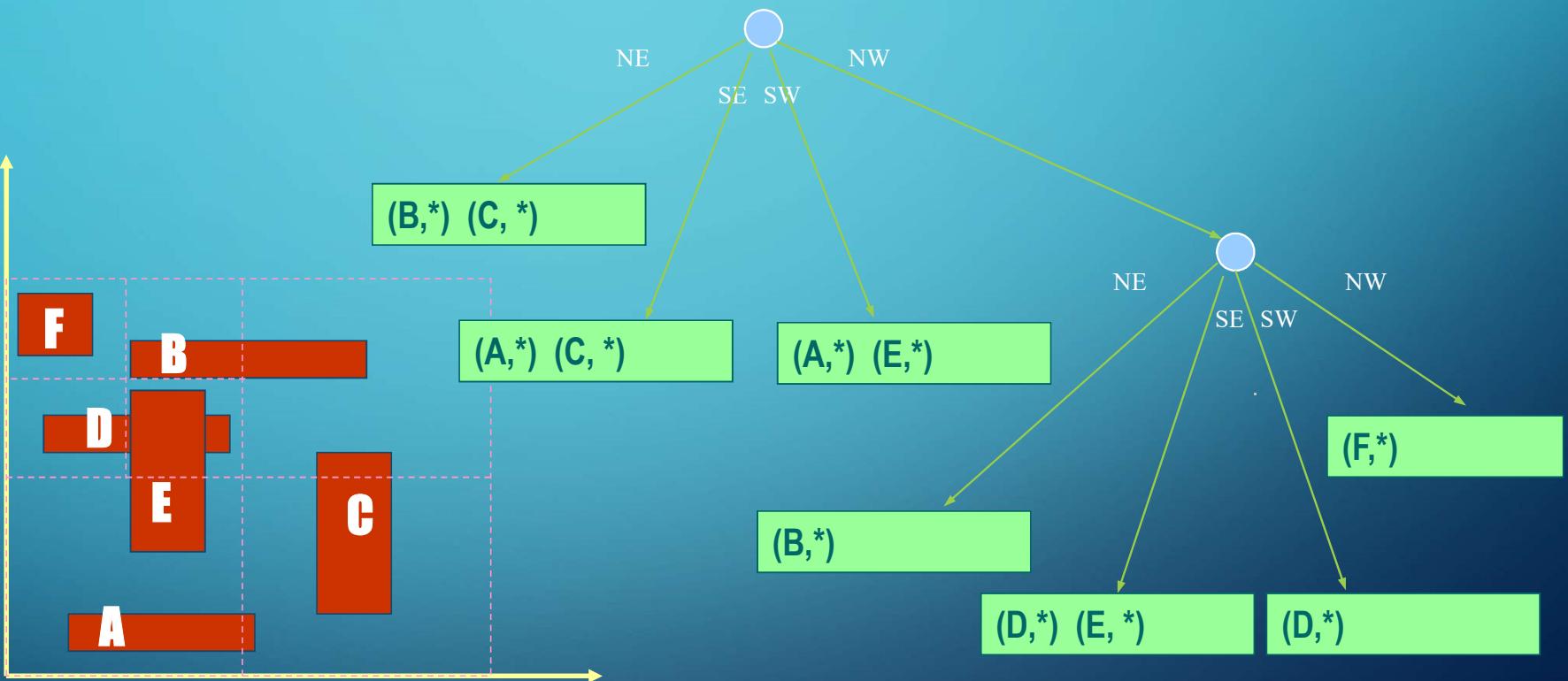
QUADTREE PARA GIS



QUADTREE PARA GIS



QUADTREE PARA GIS





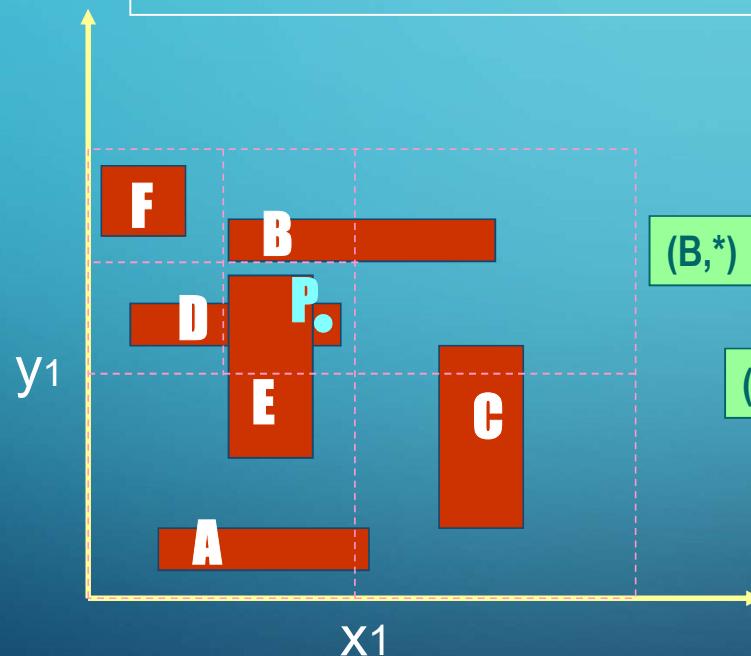
COMO CONSULTAR SOBRE QUADTREE

Las búsquedas por igualdad y rangos de uno o más campos se hacen recorriendo las ramas que abarcan el/los rango/s deseados.

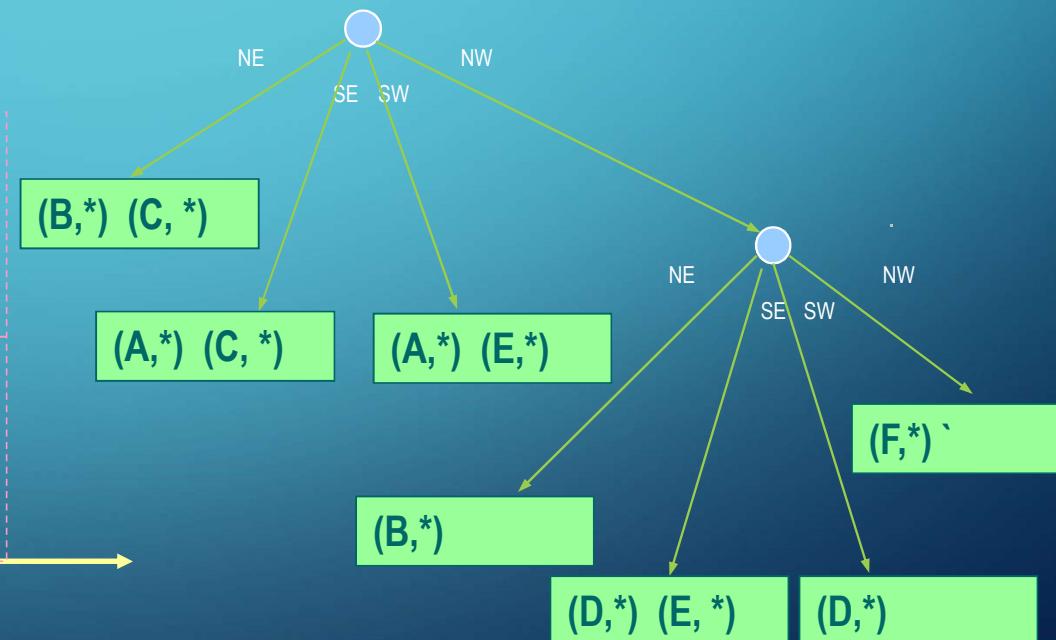
Al llegar a una hoja, se busca secuencialmente en el bucket allí presente.

CONSULTA PUNTUAL

Select X.rectangulo
where $P \in X$

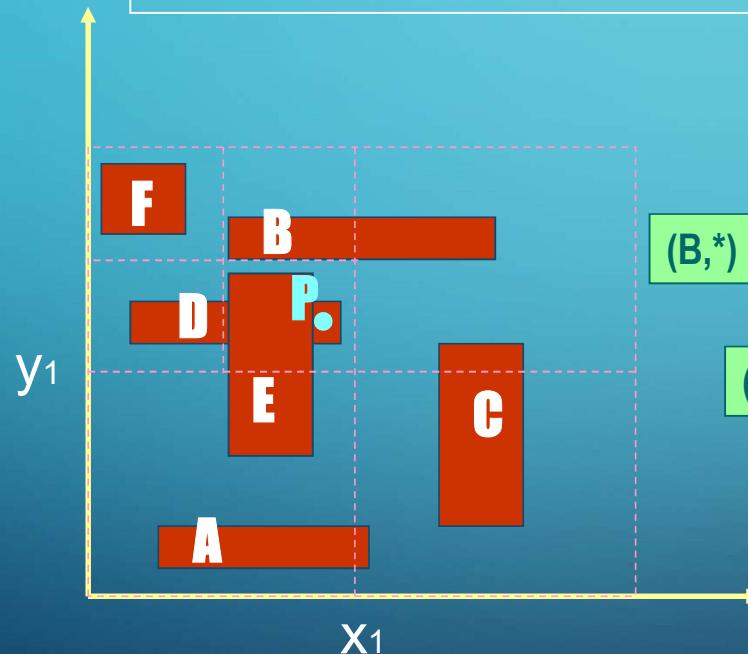


$P(x_p, y_p) ?$



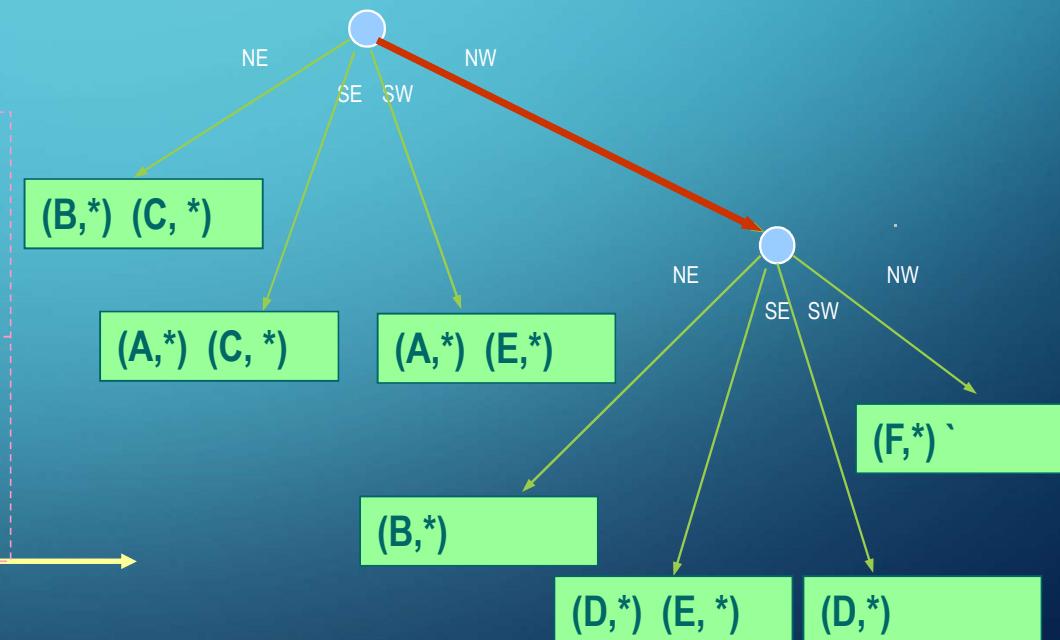
CONSULTA PUNTUAL

Select X.rectangulo
where $P \in X$



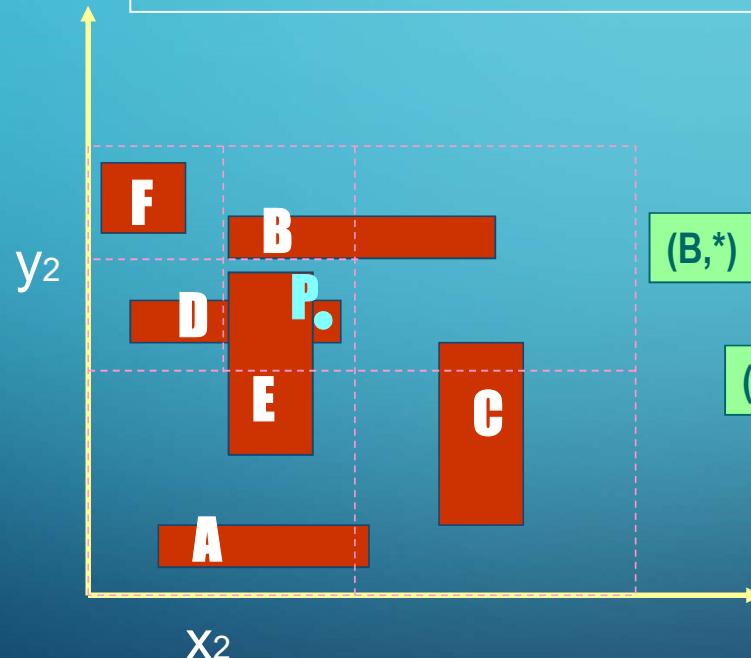
$P(x_p, y_p) ?$

$x_p < x_1 \text{ AND } y_p > y_1 \Rightarrow \text{NW}$



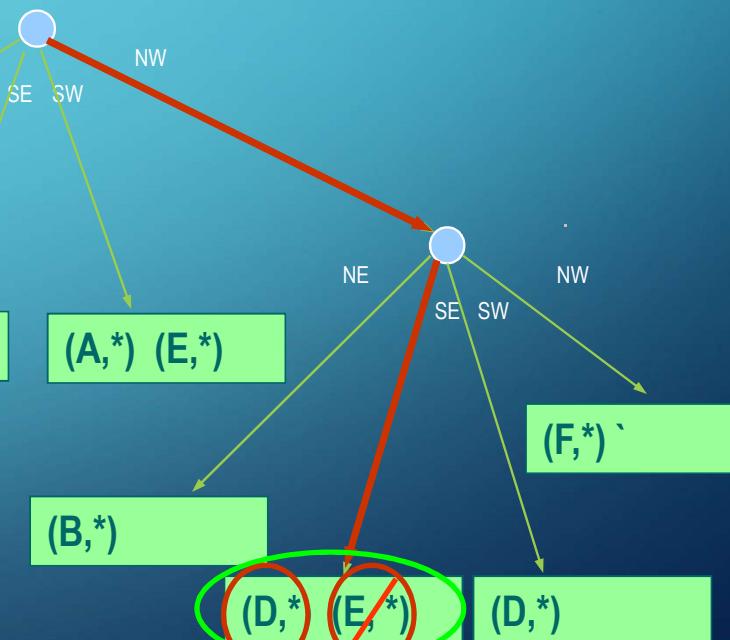
CONSULTA PUNTUAL

Select X.rectangulo
where $P \in X$



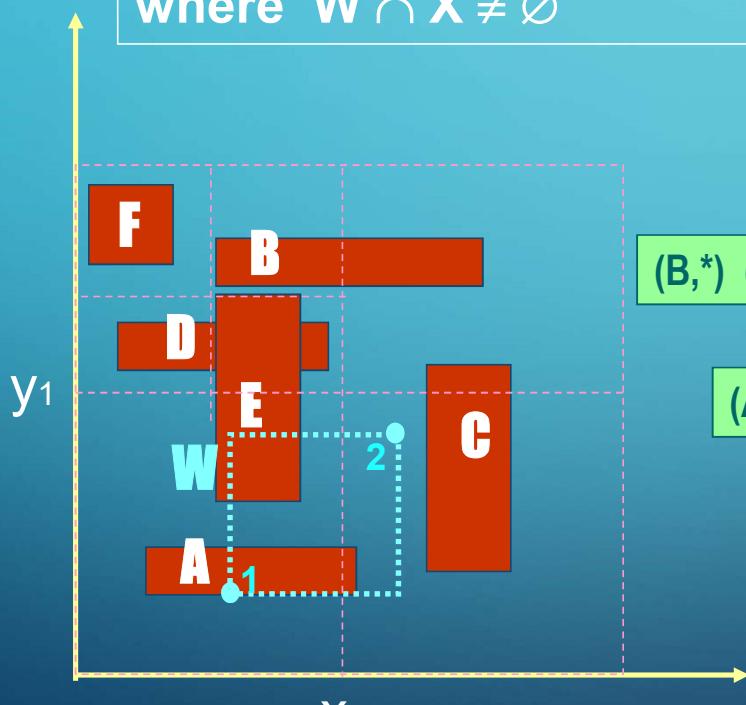
$P(x_p, y_p) ?$

$x_p > X_2 \text{ AND } y_p < y_2 \Rightarrow \text{SE}$



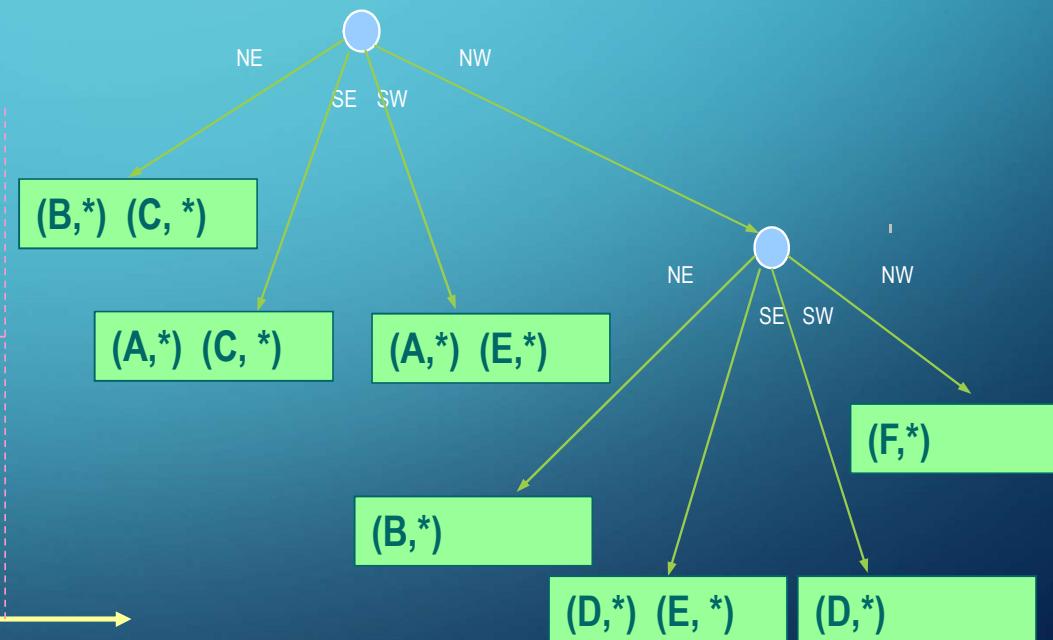
CONSULTA VENTANA

Select X.rectangulo
where $W \cap X \neq \emptyset$



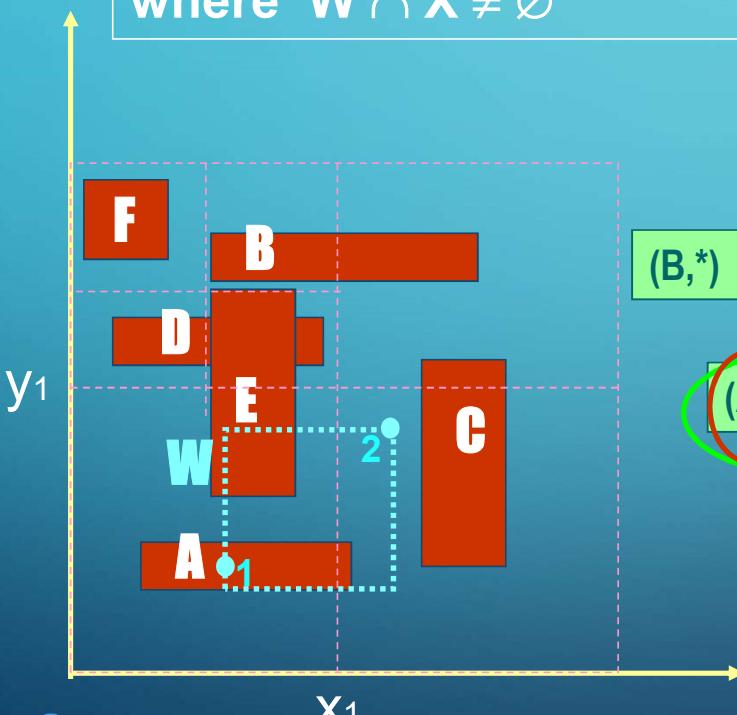
$W(x_{w1}y_{w1},x_{w2}y_{w2}) ?$

$x_{w1} < x_1 \text{ AND } y_{w1} < y_1 \Rightarrow SW$
 $x_{w2} > x_1 \text{ AND } y_{w2} < y_1 \Rightarrow SE +$



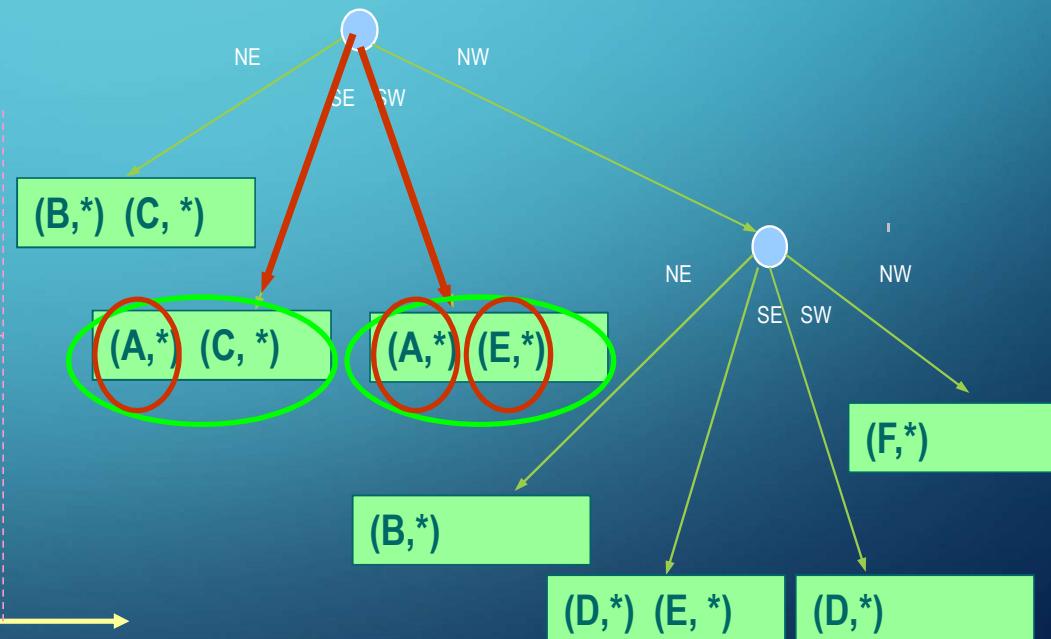
CONSULTA VENTANA

Select X.rectangulo
where $W \cap X \neq \emptyset$



W ($x_{w1}y_{w1},x_{w2}y_{w2}$) ?

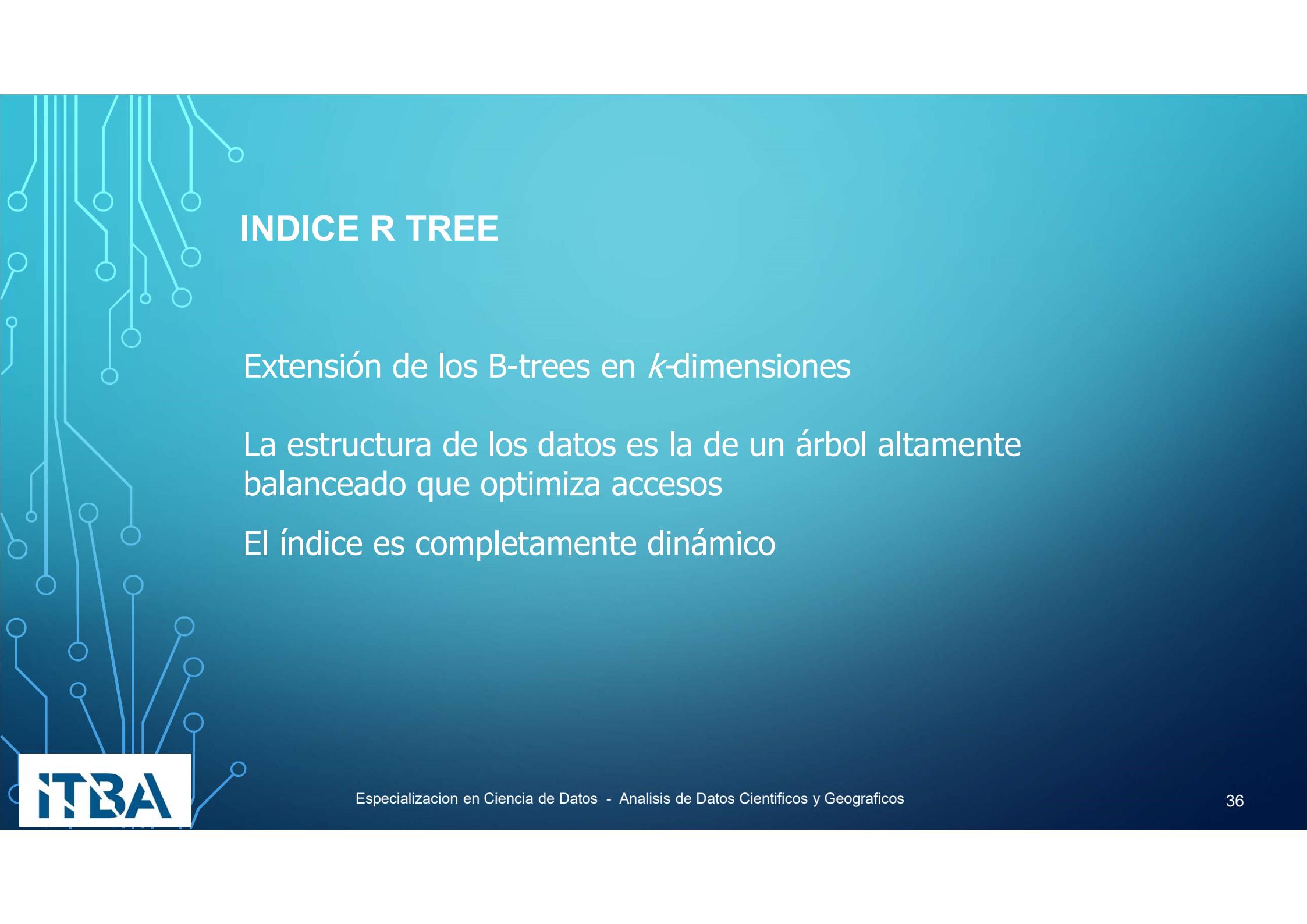
$x_{w1} < x_1 \text{ AND } y_{w1} < y_1 \Rightarrow \text{SW} +$
 $x_{w2} > x_1 \text{ AND } y_{w2} < y_1 \Rightarrow \text{SE} +$





CONCLUSIONES SOBRE QUADTREE

- Buena performance para todo tipo de consultas 
- Permite un almacenamiento eficiente de puntos no uniformemente distribuidos 
- Dividir los rangos siempre en la mitad permite ahorrar espacio en los nodos 
- Las búsquedas por proximidad son bastante eficientes 
- Puede haber muchos punteros nulos si los puntos no están bien dispersos 



INDICE R TREE

Extensión de los B-trees en k -dimensiones

La estructura de los datos es la de un árbol altamente balanceado que optimiza accesos

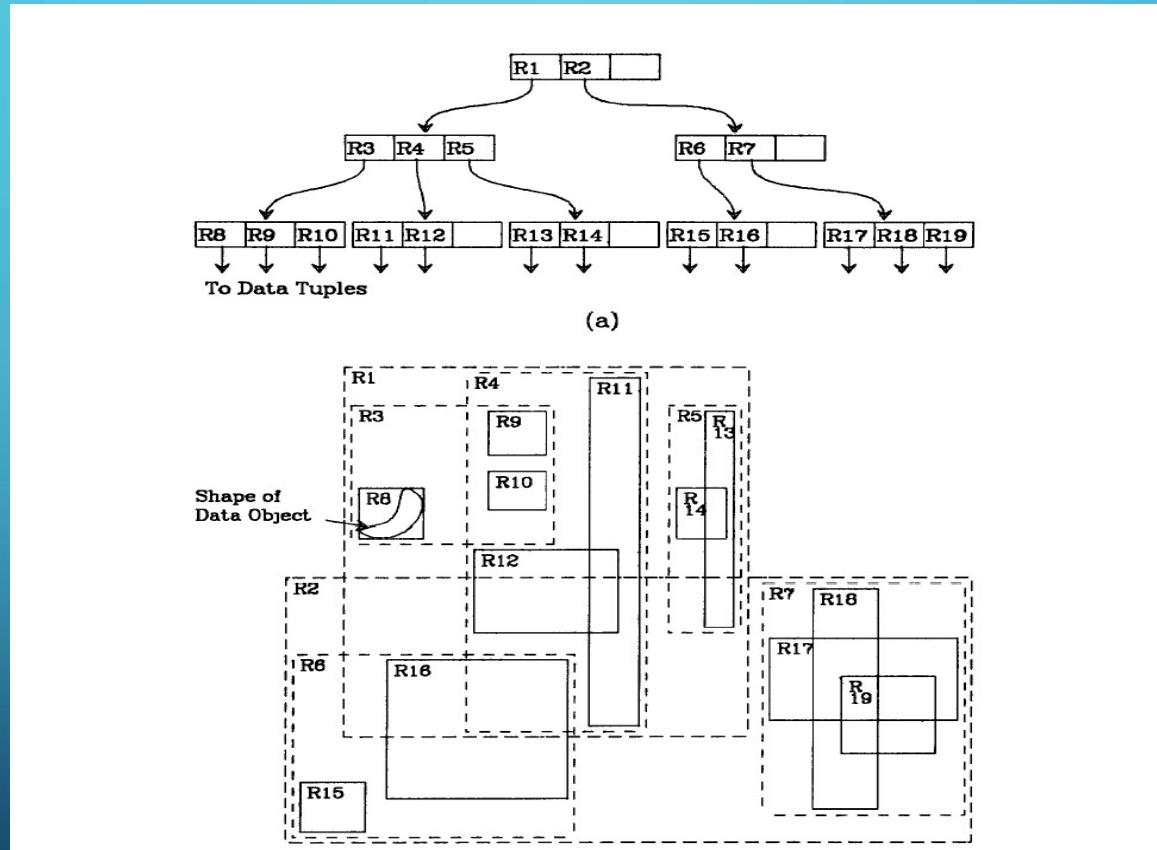
El índice es completamente dinámico



INDICE R TREE

- Nodos internos: (mbb, puntero_a_nodo) donde cada MBR encierra los MBR de sus hijos
- Nodos hojas: (mbb, rid)
- Cada nodo tiene máximo M entradas, mínimo **M/2** (excepto la raíz)
- Todas las hojas están en el mismo nivel
- Altura: $| \log_m N | - 1$

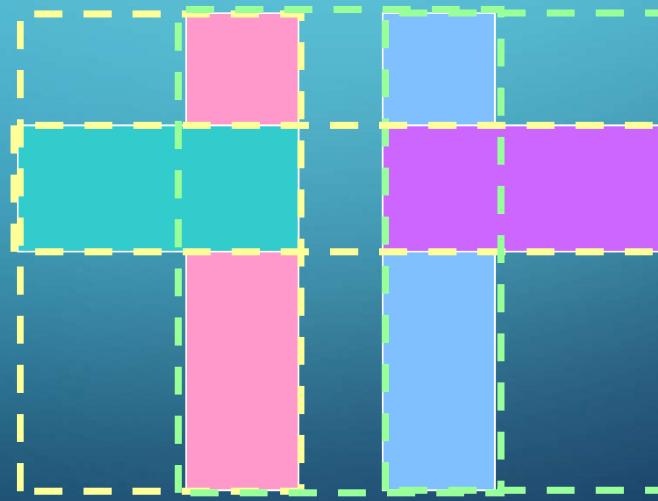
INDICE R TREE



CREACION DE R TREE

Que criterio seguir para dividir un nodo ?

- Optimizar el área cubierta por un nodo
- Optmizar la superposición de nodos





CREACION DE R TREE

R Tree busca minimizar el Area desperdiciada por cada rectangulo



Introduce mucho solapamiento



Desmejora a la hora de hacer una búsqueda....

Pensemos ... Por que ?



INDICE R+ TREE

Los R+ Tree son una variación de los R Tree que evita el solapamiento de rectángulos en los nodos intermedios.

Al evitar el solapamiento se incrementa la altura del árbol, pero el beneficio obtenido al reducir los caminos de búsquedas es mayor.

No satisfacen la propiedad de que cada hoja contiene entre $M/2$ y M entradas.



CREACION DE R TREE

- *Fuerza bruta*
- *Split cuadratico (en R Tree)*
- *Split por eje dominante (en R+ Tree)*
- *otros*



CREACION DE R TREE

- *Fuerza bruta*
- *Split cuadratico (en R Tree)*
- *Split por eje dominante (en R+ Tree)*
- *otros*



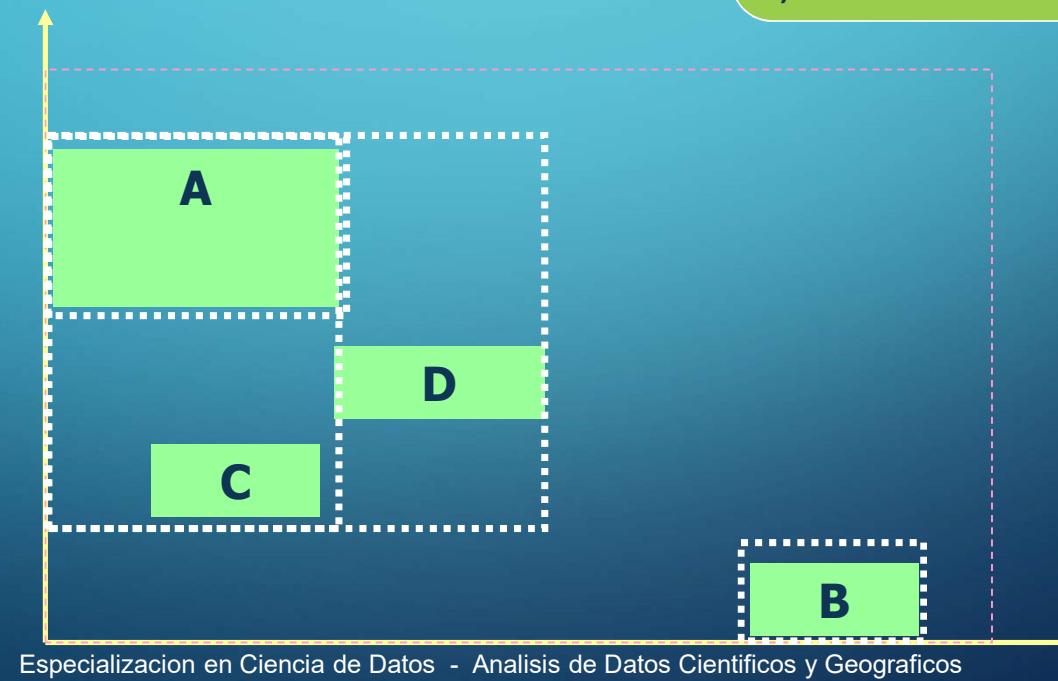
Se eligen dos objetos semilla (alejados) y se le van agregando los otros objetos al nodo, minimizando (en este orden) :

- 1) Aumento de area para incluir al nuevo objeto
- 2) Area total del nuevo rectangulo
- 3) Cantidad de integrantes

CREACION DE R TREE CON SPLIT CUADRATICO

Minimizar (en este orden) :

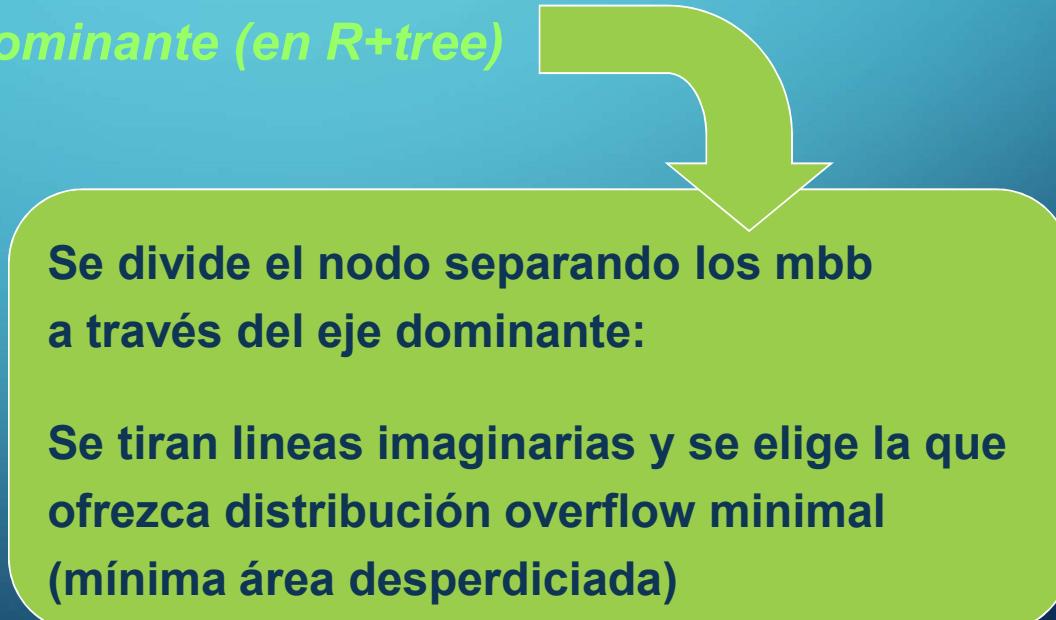
- 1) Aumento de area para incluir al nuevo objeto
- 2) Area total del nuevo rectangulo
- 3) Cantidad de integrantes





CREACION DE R TREE

- *Fuerza bruta*
- *Split cuadratico (en R -Tree)*
- *Split por eje dominante (en R+tree)*
- *otros*

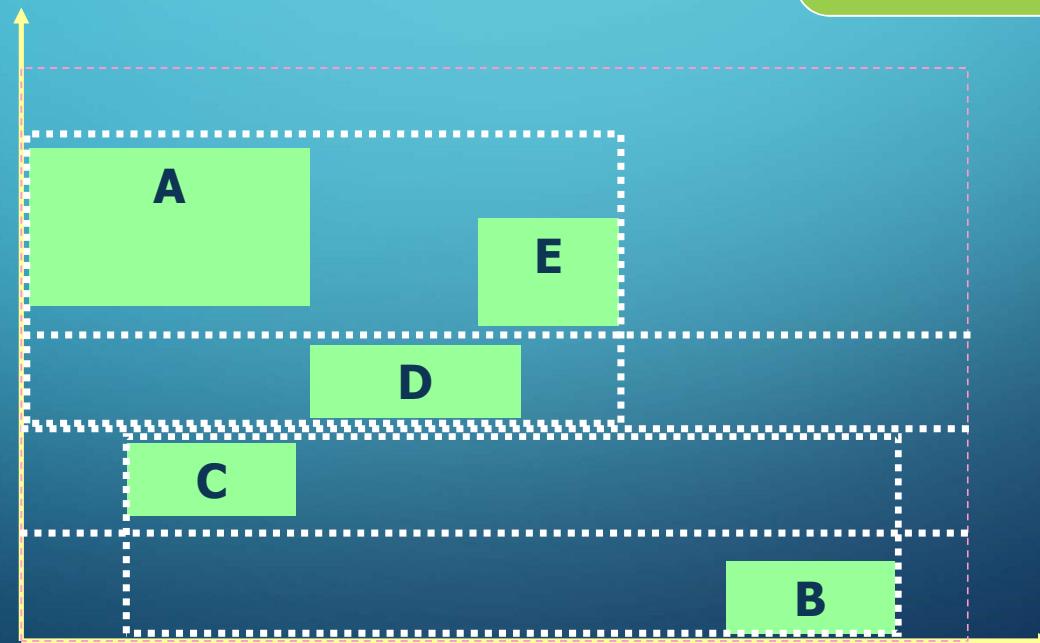


Se divide el nodo separando los mbb
a través del eje dominante:

Se tiran líneas imaginarias y se elige la que
ofrezca distribución overflow minimal
(mínima área desperdiciada)

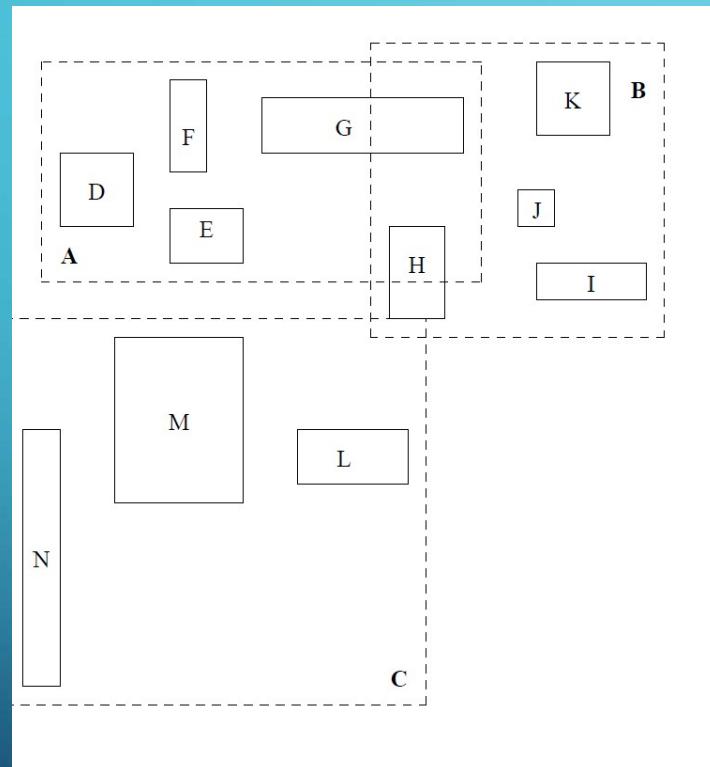
CREACION DE R TREE CON SPLIT EJE DOMINANTE

Se tiran líneas imaginarias y se elige la que ofrezca distribución overflow minimal (mínima área desperdiciada)



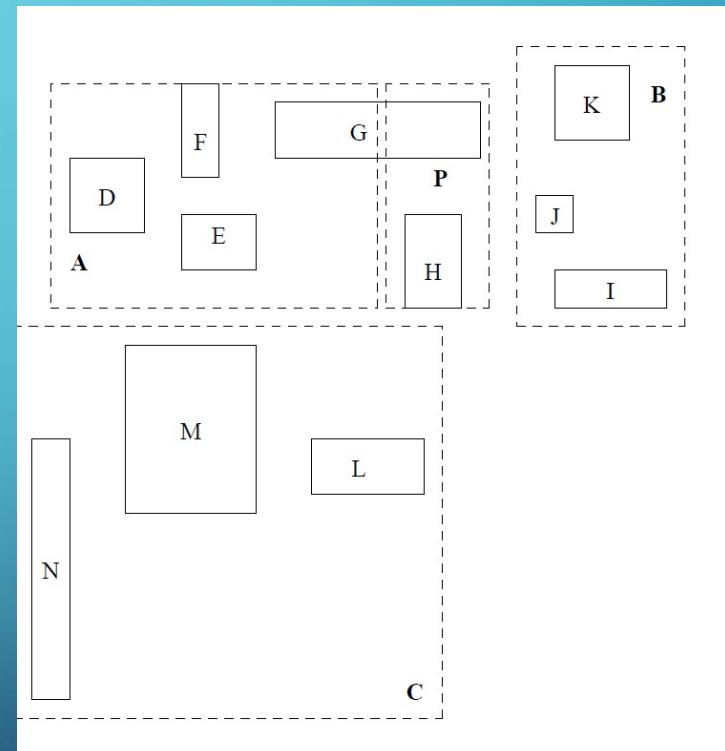


R TREE



vs.

R+ TREE



EJEMPLO CREACION DE R+ TREE

Zonas				
X1	Y1	X2	Y2	id
0	500	30	450	01
70	480	100	420	02
100	500	180	450	03
100	420	120	400	04
200	500	250	450	05
300	500	330	470	06
370	500	400	450	07
450	500	500	450	08
0	400	20	380	04
50	380	100	350	09
100	340	140	300	10
180	380	200	300	11
200	380	250	310	02
280	380	300	320	06
400	400	430	350	12
50	300	100	250	06
200	300	300	200	08
300	300	320	280	04

Suponemos bloques de disco con capacidad para 3 tuplas

id	Descripción
01	Iglesia
02	Restaurant
03	Escuela
04	Quiosco
05	Comisaría
06	Confitería
07	Discoteca
08	Estacionamiento
09	Locutorio
10	Lavadero
11	Cine
12	Hotel
13	Barrio A
14	Barrio B
15	Cable VCC
16	Multicanal
17	CableVisión

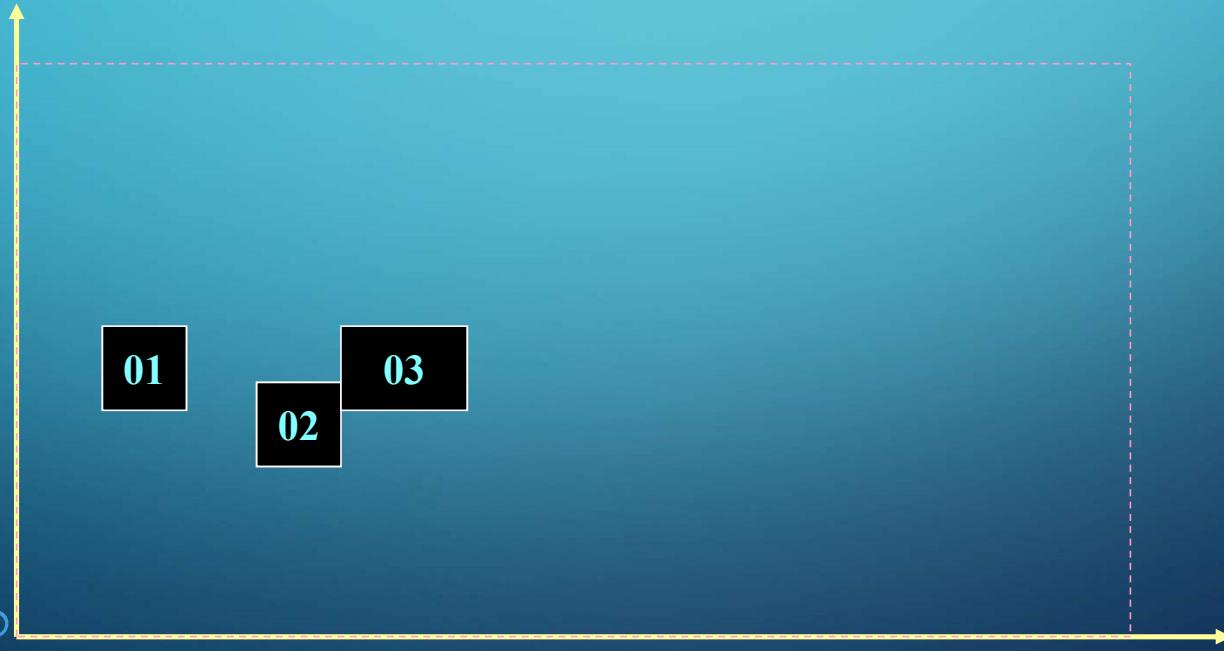


EJEMPLO DE R+ TREE

((0,30,450,500),01)
((70,100,420,480),02)
((100,180,450,500),03)



01 | 02 | 03

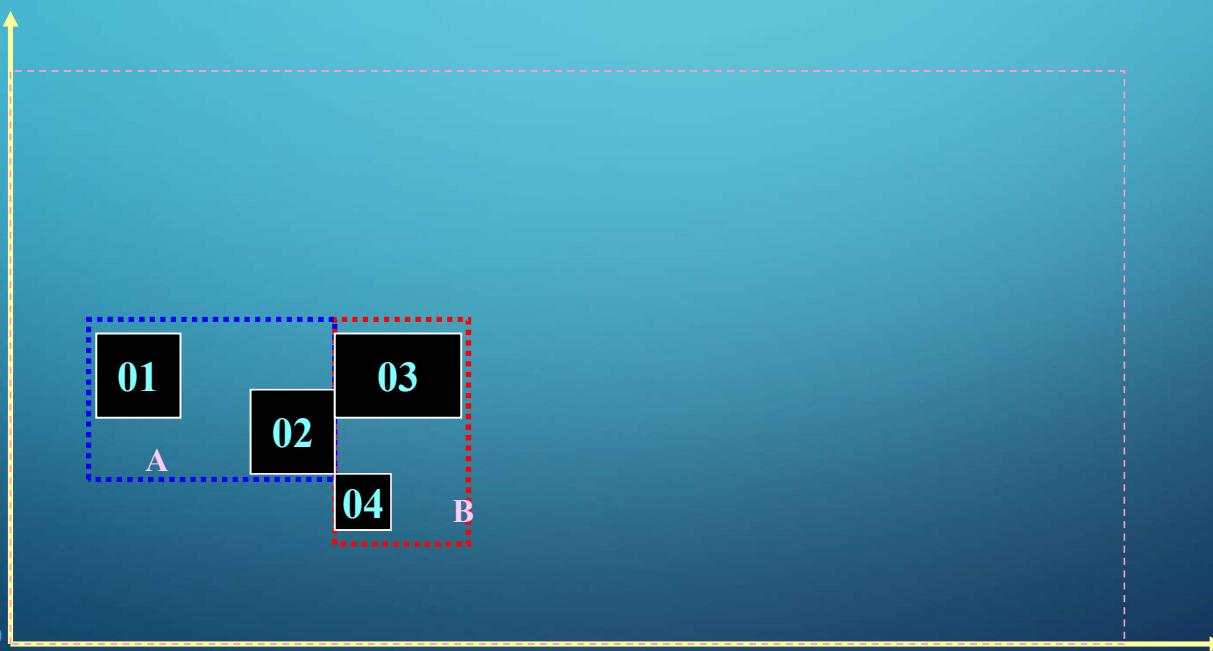


01
02
03

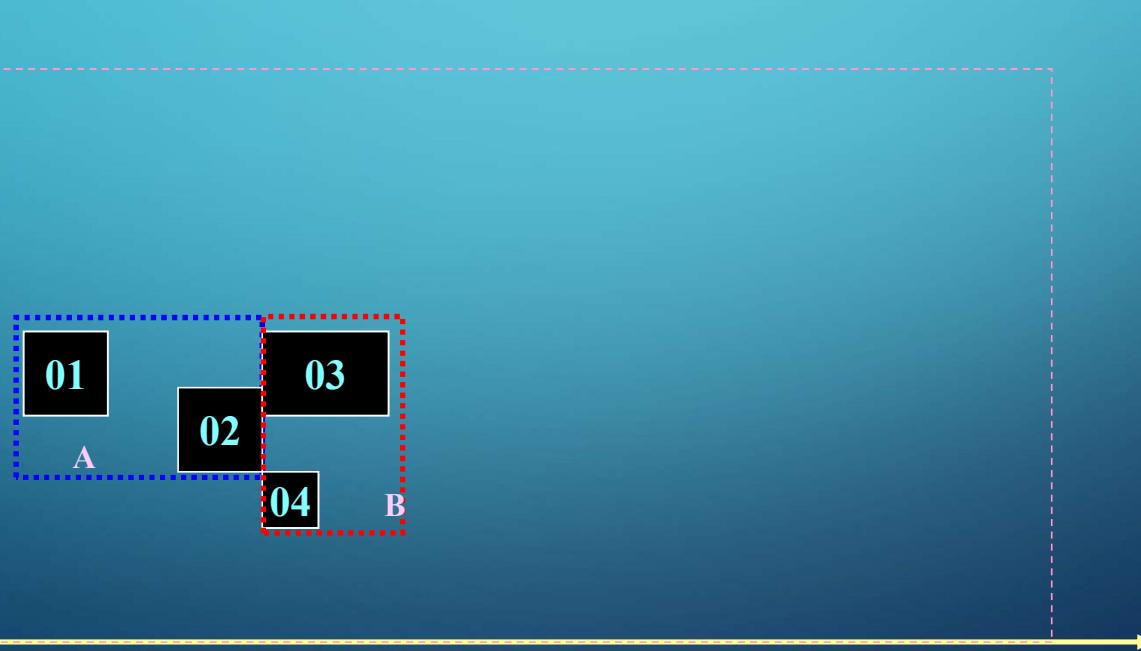
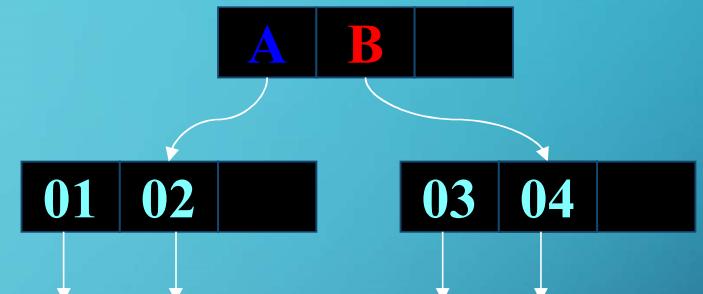
EJEMPLO DE R+ TREE

((100,120,400,420),04)

01	02	03	04
----	----	----	----

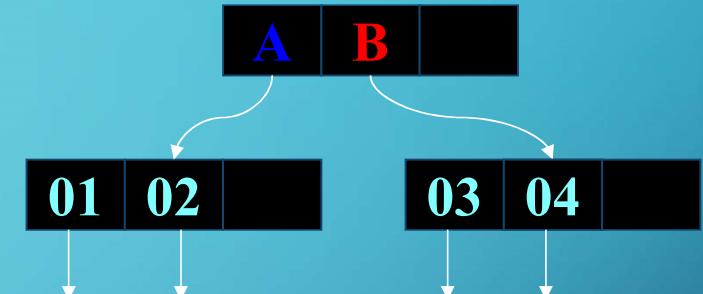
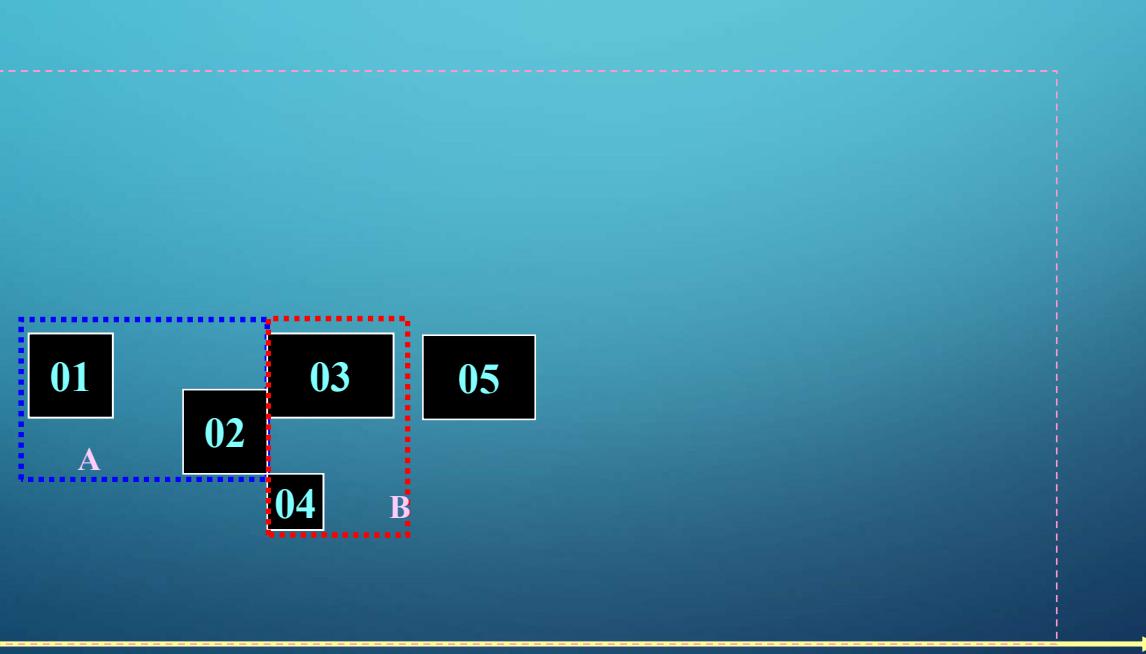


EJEMPLO DE R+ TREE



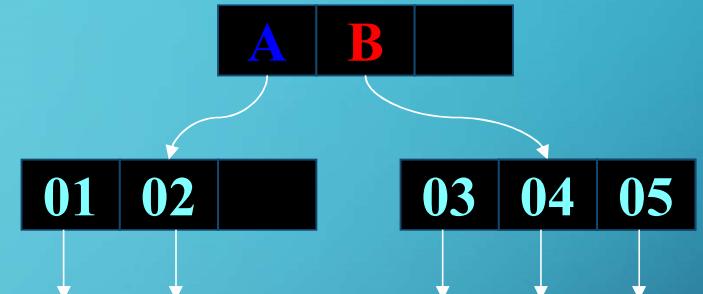
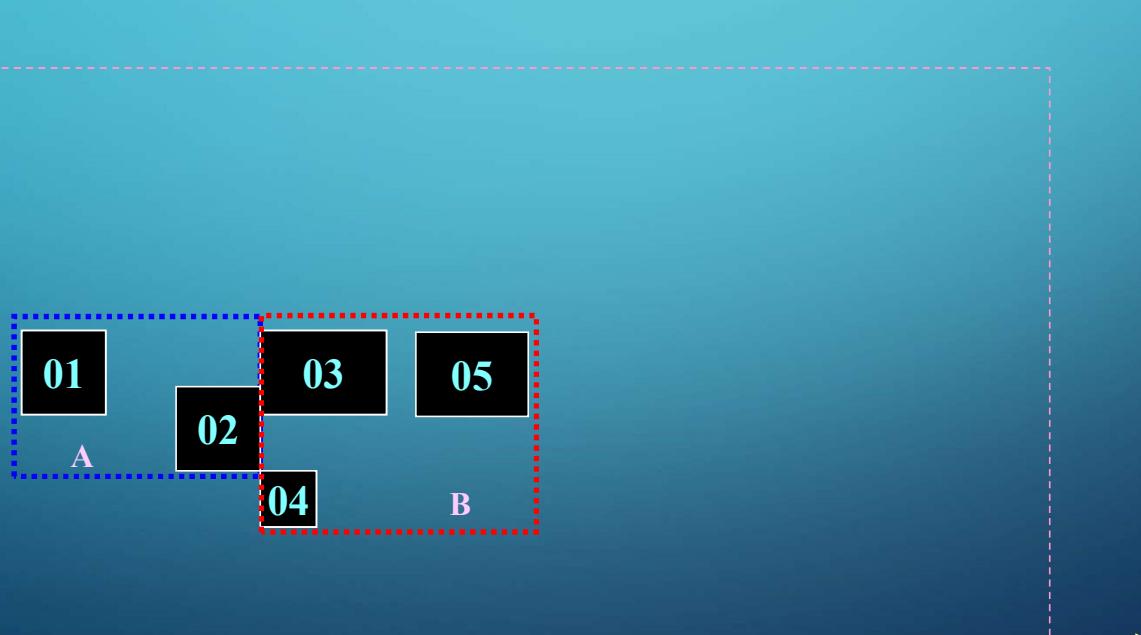
EJEMPLO DE R+ TREE

((200,250,450,500),05)



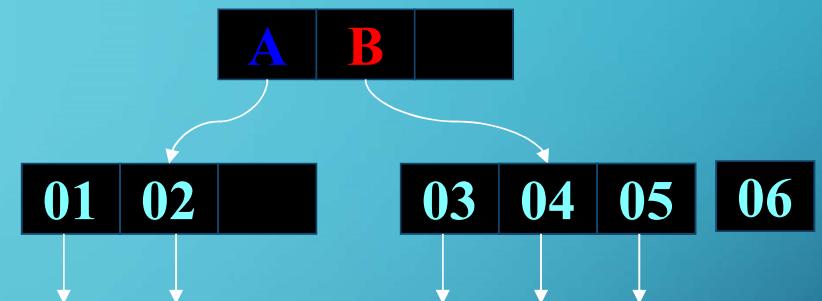
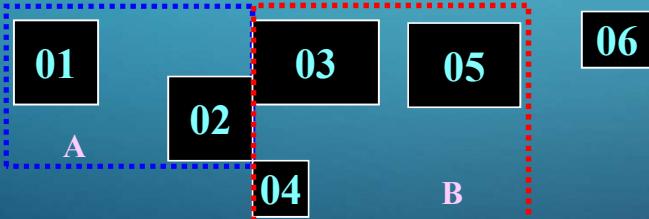
EJEMPLO DE R+ TREE

((200,250,450,500),05)



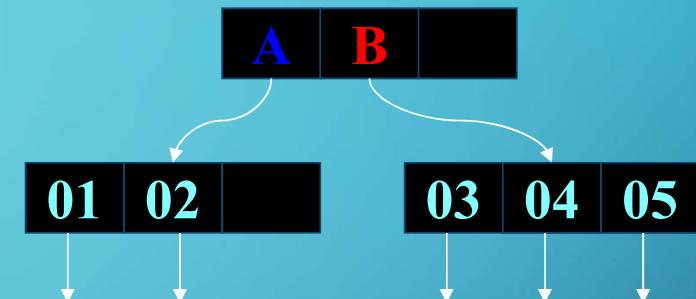
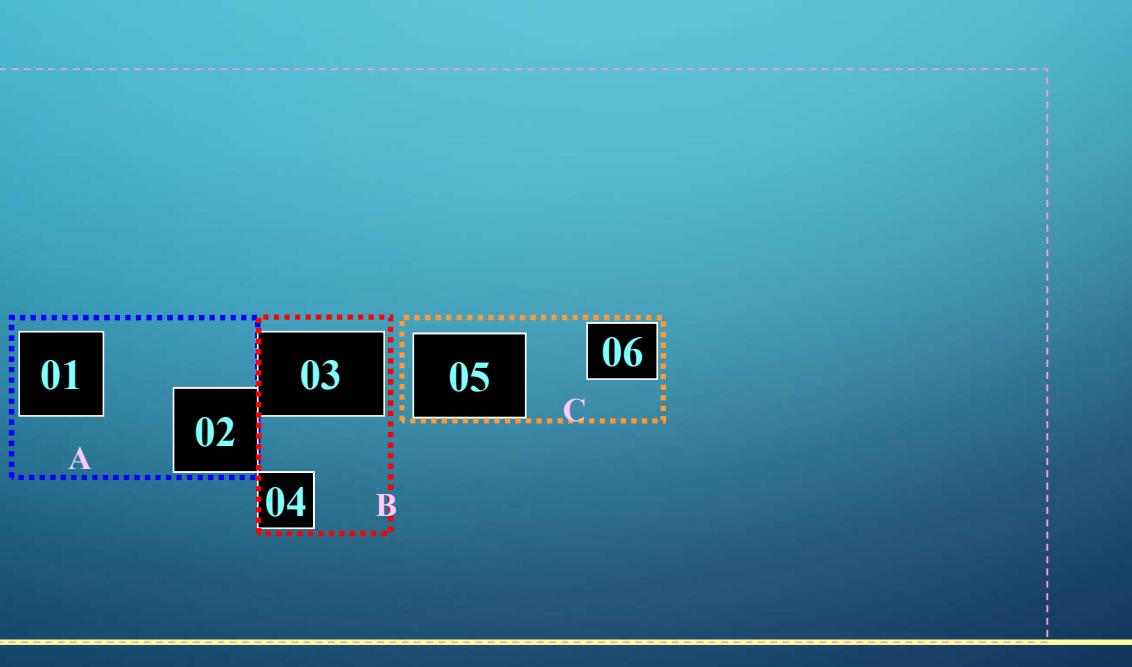
EJEMPLO DE R+ TREE

((300,330,470,500),06)



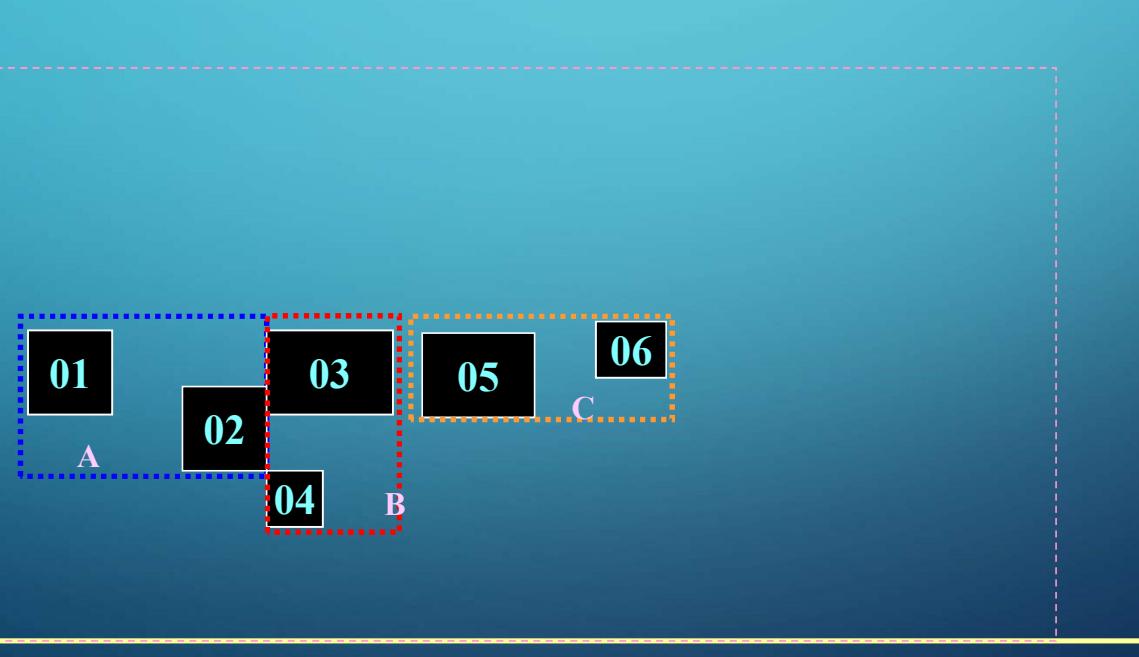
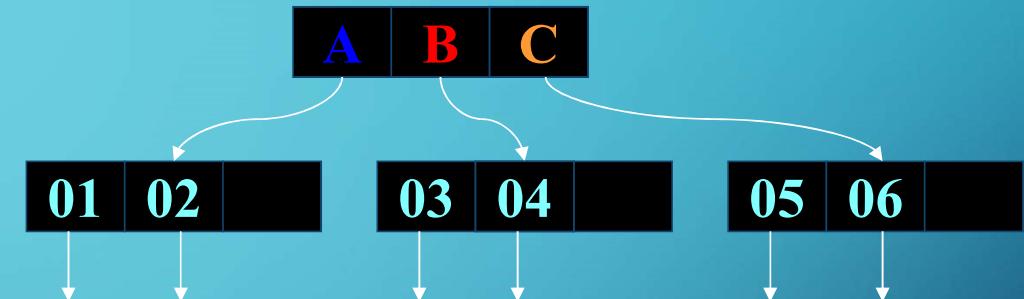
EJEMPLO DE R+ TREE

((300,330,470,500),06)



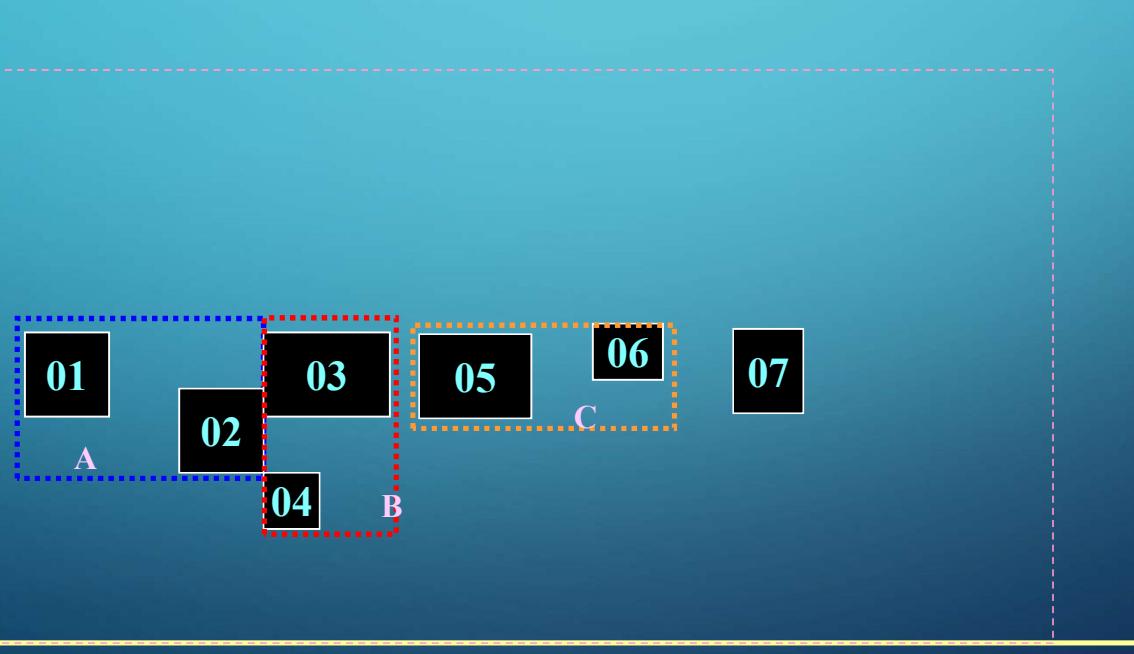
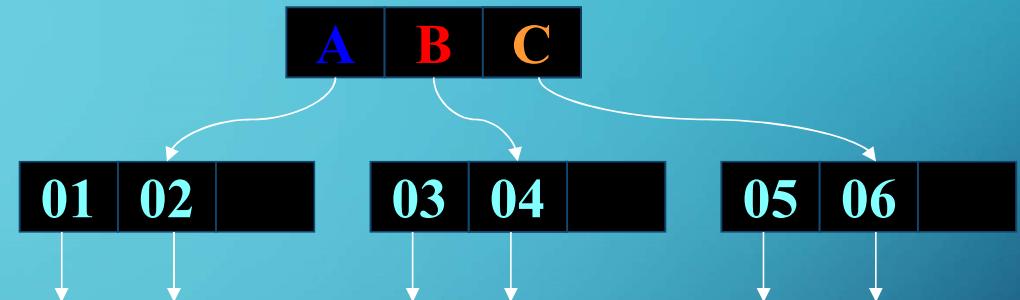
EJEMPLO DE R+ TREE

((300,330,470,500),06)



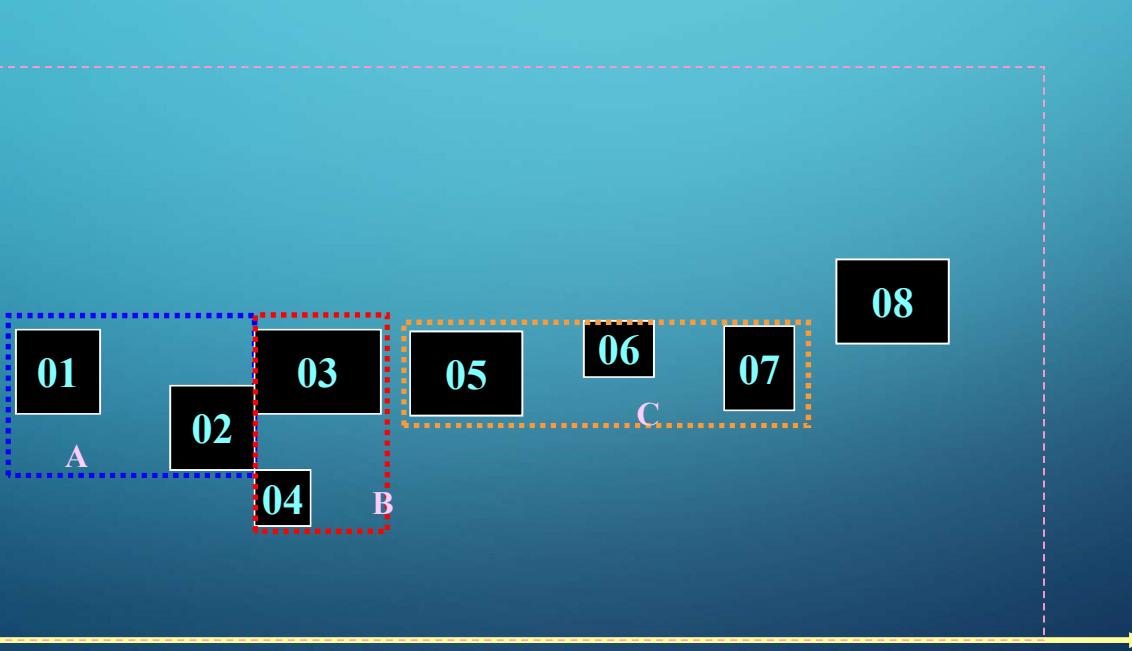
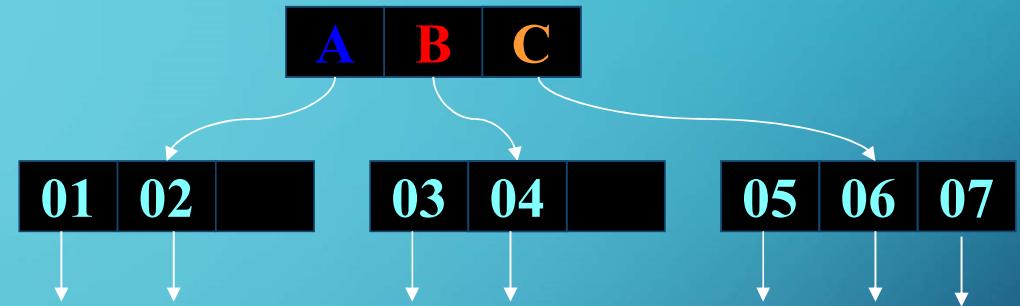
EJEMPLO DE R+ TREE

((370,400,450,500),07)



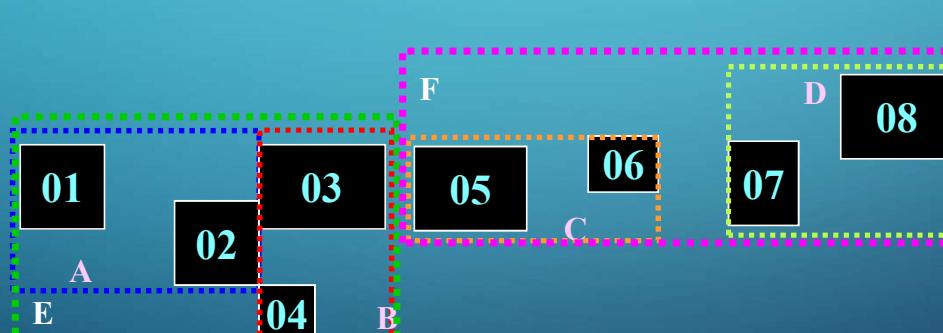
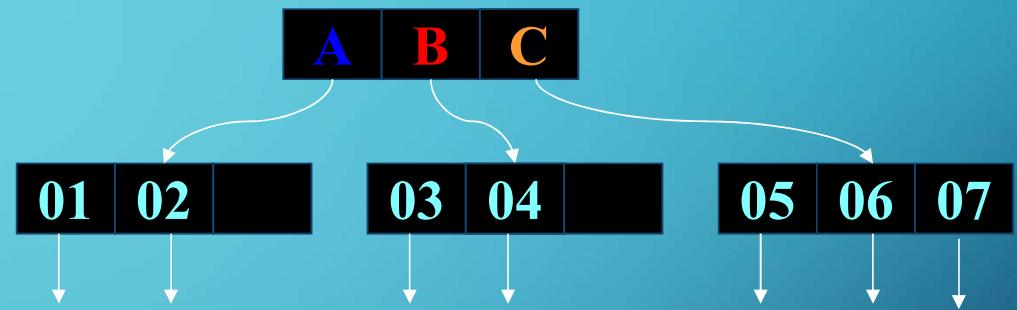
EJEMPLO DE R+ TREE

((450,500,450,500),08)

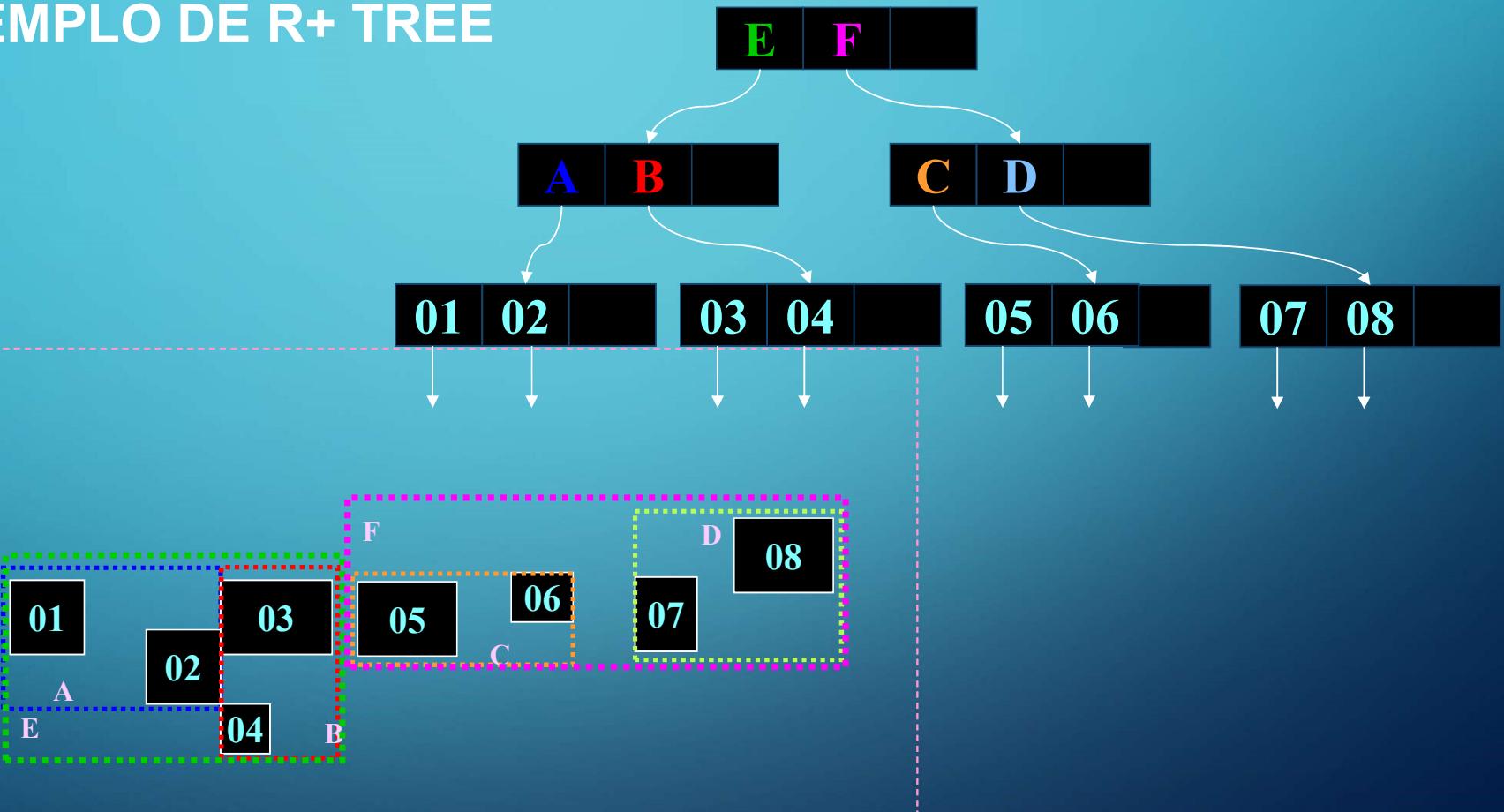


EJEMPLO DE R+ TREE

((450,500,450,500),08)



EJEMPLO DE R+ TREE



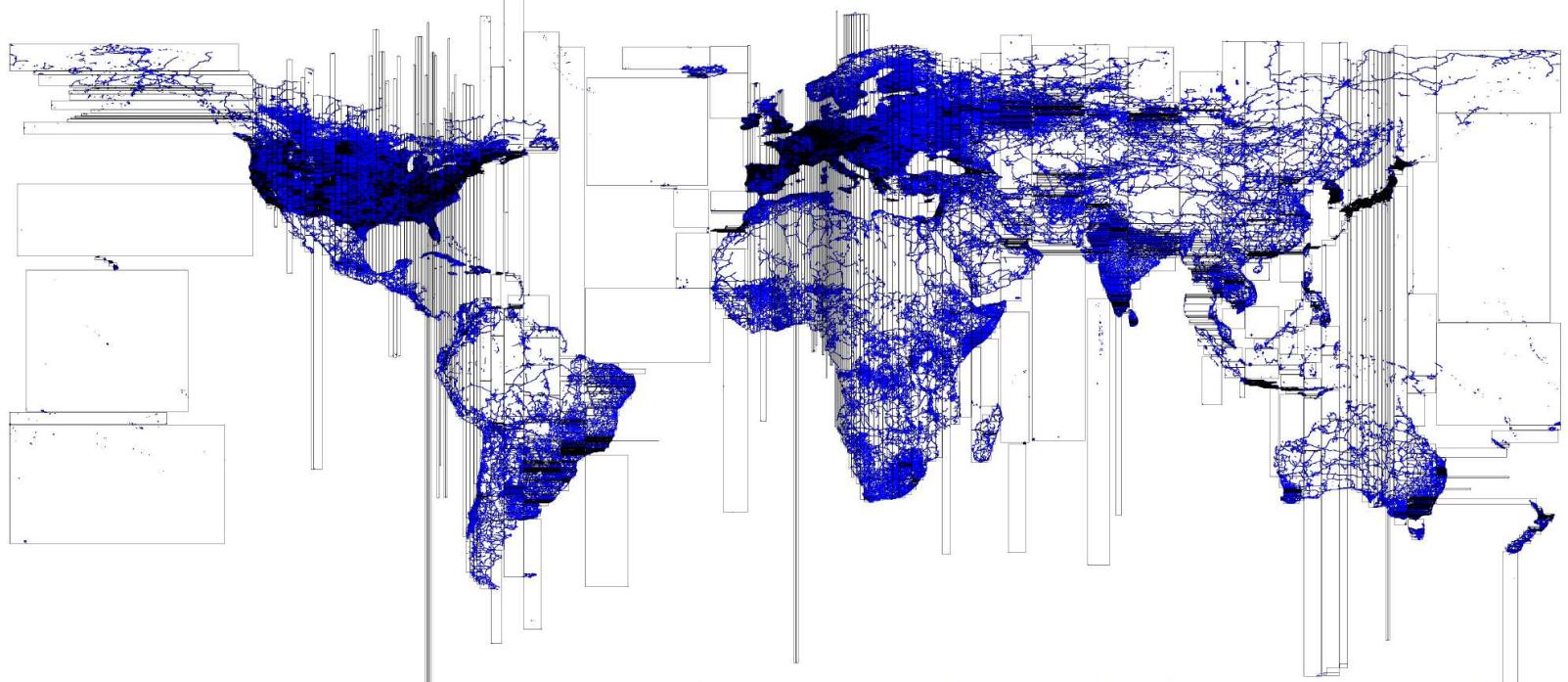
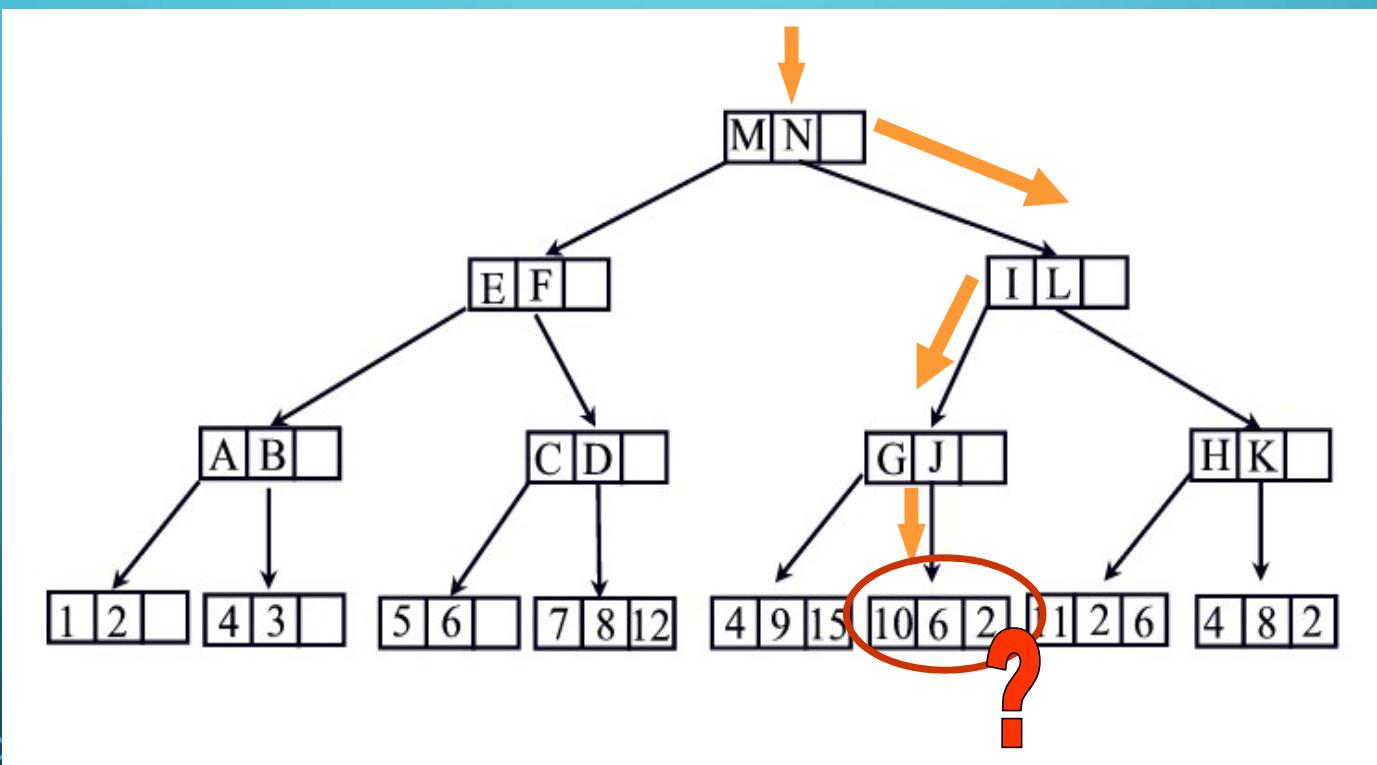


Fig. 5. R+-tree index for road segments in the whole world extracted from OpenStreetMap (Best viewed in color)

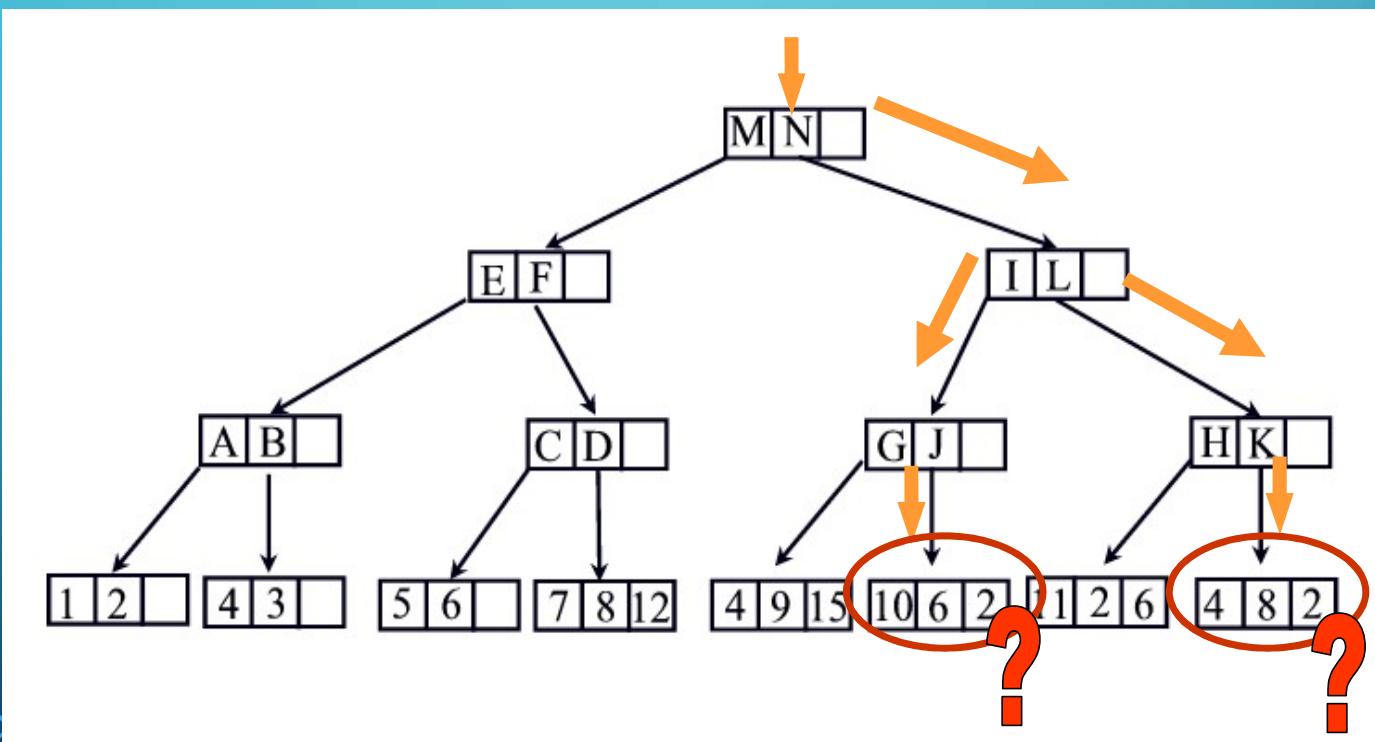
CONSULTA PUNTUAL EN R+ TREE

Se busca hasta las hojas, entrando por los nodos cuyo mbb contiene al punto



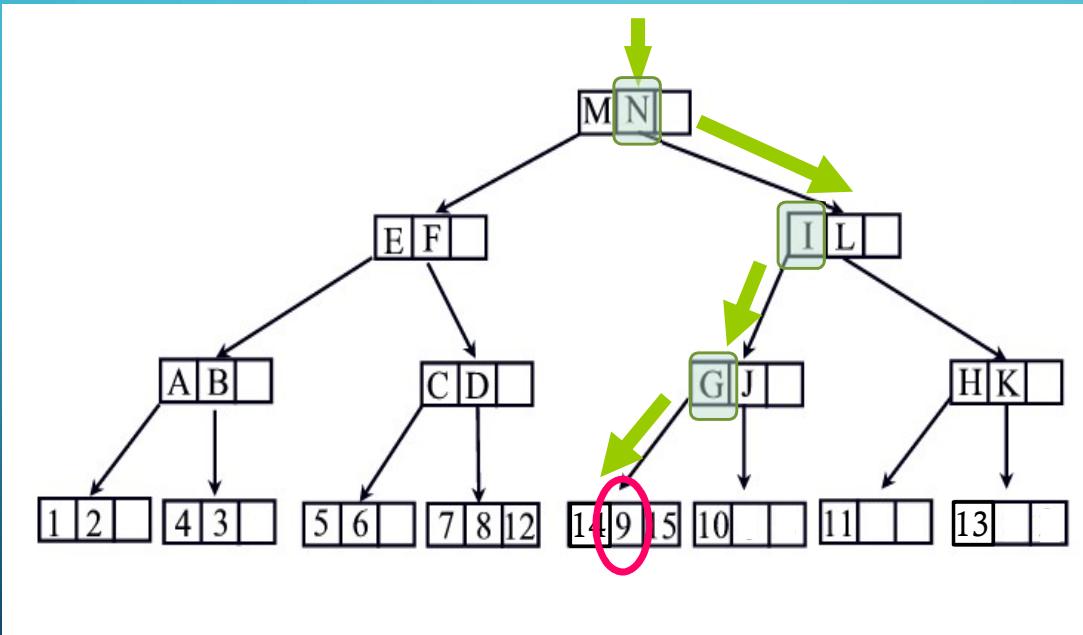
CONSULTA VENTANA EN R+ TREE

Se busca hasta las hojas, entrando por los nodos cuyo MBR intersecta a la ventana



EJEMPLO BUSQUEDA PUNTUAL

Objeto cuya superficie contiene al punto de coordenadas (110, 150)

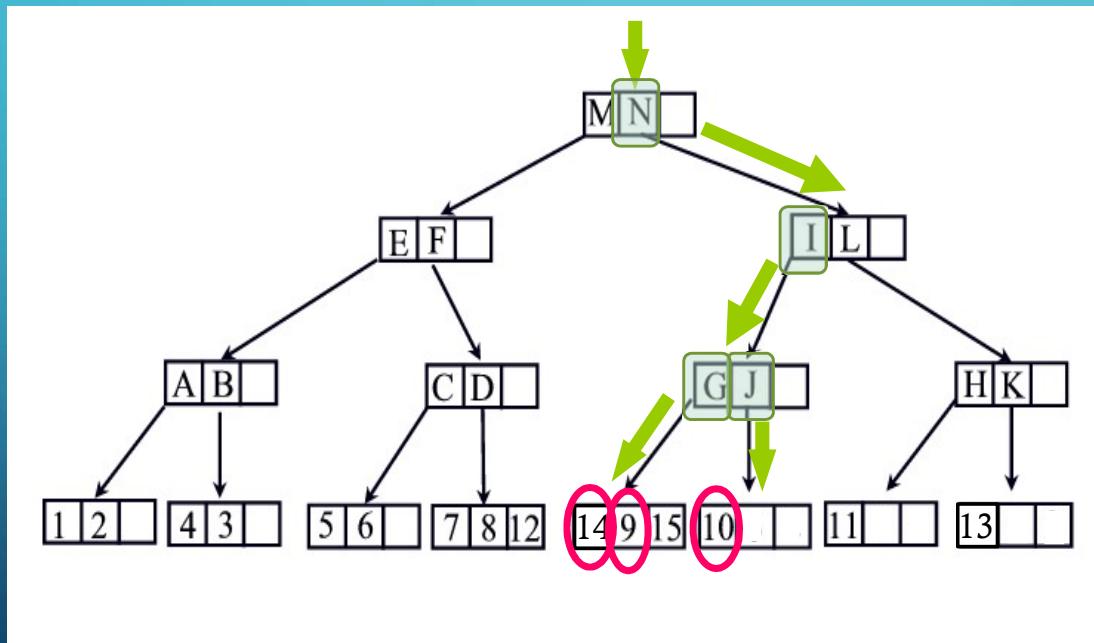


Nodos	
Nombre	(X1, X2, Y1, Y2)
A	(0, 100, 400, 500)
B	(100, 200, 350, 500)
C	(200, 300, 450, 500)
D	(200, 350, 250, 450)
E	(0, 200, 350, 500)
F	(200, 350, 250, 500)
G	(0, 200, 100, 200)
H	(300, 400, 50, 180)
I	(0, 200, 0, 200)
J	(80, 200, 0, 100)
K	(400, 500, 0, 170)
L	(300, 500, 0, 180)
M	(0, 350, 250, 500)
N	(0, 500, 0, 200)

Hojas	
Hojas	(X1, X2, Y1, Y2)
1	(0, 50, 480, 500)
2	(55, 100, 400, 450)
3	(100, 120, 400, 500)
4	(180, 200, 350, 390)
5	(200, 220, 450, 470)
6	(250, 300, 480, 500)
7	(200, 210, 370, 450)
8	(220, 260, 250, 280)
9	(100, 120, 130, 180)
10	(80, 200, 0, 100)
11	(300, 400, 50, 180)
12	(280, 350, 300, 350)
13	(400, 500, 0, 170)
14	(0, 100, 100, 120)
15	(170, 200, 190, 200)

EJEMPLO BUSQUEDA VENTANA

Objetos que se encuentran dentro de la zona definida por $0 \leq x \leq 290, 0 \leq y \leq 190$

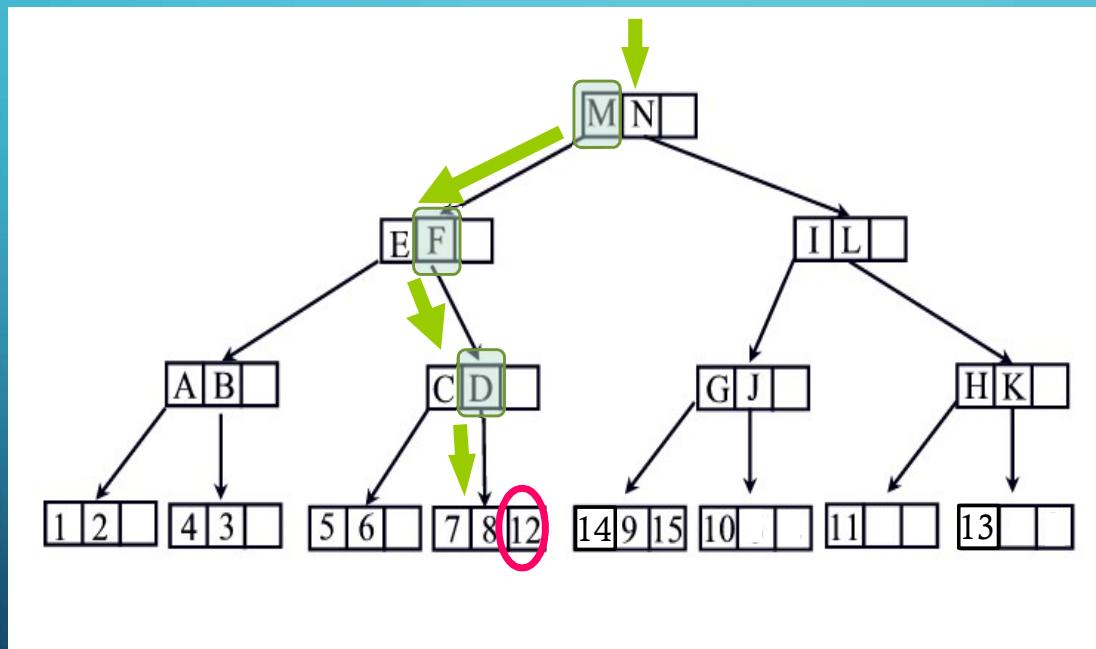


Nodos	
Nombre	(X1, X2, Y1, Y2)
A	(0, 100, 400, 500)
B	(100, 200, 350, 500)
C	(200, 300, 450, 500)
D	(200, 350, 250, 450)
E	(0, 200, 350, 500)
F	(200, 350, 250, 500)
G	(0, 200, 100, 200)
H	(300, 400, 50, 180)
I	(0, 200, 0, 200)
J	(80, 200, 0, 100)
K	(400, 500, 0, 170)
L	(300, 500, 0, 180)
M	(0, 350, 250, 500)
N	(0, 500, 0, 200)

Hoja	
Hoja	(X1, X2, Y1, Y2)
1	(0, 50, 480, 500)
2	(55, 100, 400, 450)
3	(100, 120, 400, 500)
4	(180, 200, 350, 390)
5	(200, 220, 450, 470)
6	(250, 300, 480, 500)
7	(200, 210, 370, 450)
8	(220, 260, 250, 280)
9	(100, 120, 130, 180)
10	(80, 200, 0, 100)
11	(300, 400, 50, 180)
12	(280, 350, 300, 350)
13	(400, 500, 0, 170)
14	(0, 100, 100, 120)
15	(170, 200, 190, 200)

EJEMPLO BUSQUEDA VENTANA

Objetos que cubren toda la zona definida por $300 \leq x \leq 350, 300 \leq y \leq 350$

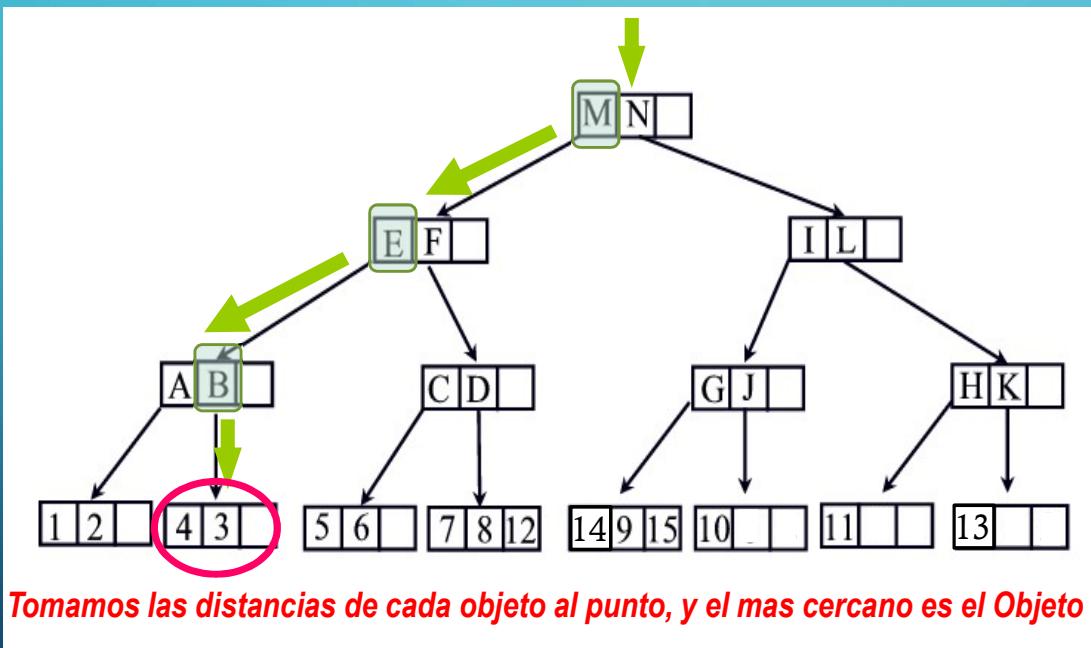


Nodos	
Nombre	(X1, X2, Y1, Y2)
A	(0, 100, 400, 500)
B	(100, 200, 350, 500)
C	(200, 300, 450, 500)
D	(200, 350, 250, 450)
E	(0, 200, 350, 500)
F	(200, 350, 250, 500)
G	(0, 200, 100, 200)
H	(300, 400, 50, 180)
I	(0, 200, 0, 200)
J	(80, 200, 0, 100)
K	(400, 500, 0, 170)
L	(300, 500, 0, 180)
M	(0, 350, 250, 500)
N	(0, 500, 0, 200)

Hoja	
Hoja	(X1, X2, Y1, Y2)
1	(0, 50, 480, 500)
2	(55, 100, 400, 450)
3	(100, 120, 400, 500)
4	(180, 200, 350, 390)
5	(200, 220, 450, 470)
6	(250, 300, 480, 500)
7	(200, 210, 370, 450)
8	(220, 260, 250, 280)
9	(100, 120, 130, 180)
10	(80, 200, 0, 100)
11	(300, 400, 50, 180)
12	(280, 350, 300, 350)
13	(400, 500, 0, 170)
14	(0, 100, 100, 120)
15	(170, 200, 190, 200)

EJEMPLO BUSQUEDA DISTANCIA

Objeto mas cercano al punto de coordenadas (110, 360)



Nodos	
Nombre	(X1, X2, Y1, Y2)
A	(0, 100, 400, 500)
B	(100, 200, 350, 500)
C	(200, 300, 450, 500)
D	(200, 350, 250, 450)
E	(0, 200, 350, 500)
F	(200, 350, 250, 500)
G	(0, 200, 100, 200)
H	(300, 400, 50, 180)
I	(0, 200, 0, 200)
J	(80, 200, 0, 100)
K	(400, 500, 0, 170)
L	(300, 500, 0, 180)
M	(0, 350, 250, 500)
N	(0, 500, 0, 200)

Hoja	
Hoja	(X1, X2, Y1, Y2)
1	(0, 50, 480, 500)
2	(55, 100, 400, 450)
3	(100, 120, 400, 500)
4	(180, 200, 350, 390)
5	(200, 220, 450, 470)
6	(250, 300, 480, 500)
7	(200, 210, 370, 450)
8	(220, 260, 250, 280)
9	(100, 120, 130, 180)
10	(80, 200, 0, 100)
11	(300, 400, 50, 180)
12	(280, 350, 300, 350)
13	(400, 500, 0, 170)
14	(0, 100, 100, 120)
15	(170, 200, 190, 200)



CONCLUSIONES SOBRE R TREE

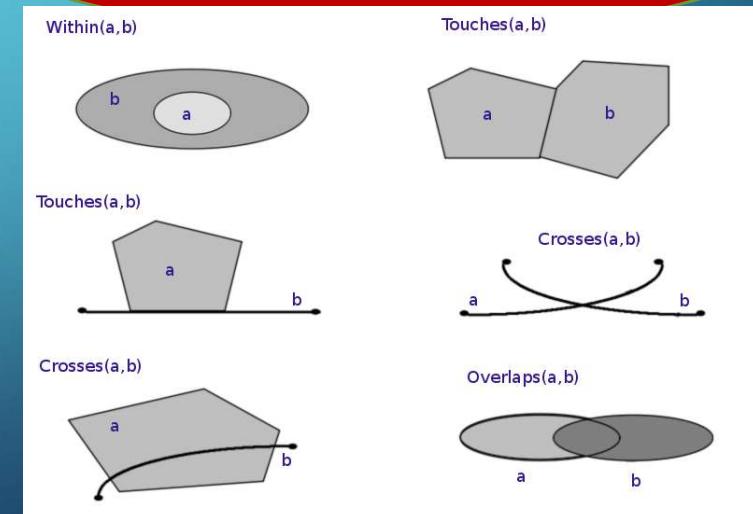
- Se pueden realizar búsquedas y borrados, sin necesidad de reordenar el árbol 
- Eficiente para búsquedas entre rangos de espacio 
- Alta eficiencia en búsquedas de objetos dentro de áreas determinadas 
- La búsqueda para un área puede conducir a recorrer varios caminos en el árbol 
- Complejidad para el cálculo del armado del árbol 

OPERADORES ESPACIALES

Solamente podemos consultar pertenencia o inclusion ?

No se pueden explotar mas profundamente los objetos y sus topologias ?

Hay muchas opciones mas para consultas espaciales !





MANOS A LA OBRA....

