

Systemes d'Information Géographique

Jean-Yves Antoine

Université François Rabelais de Tours

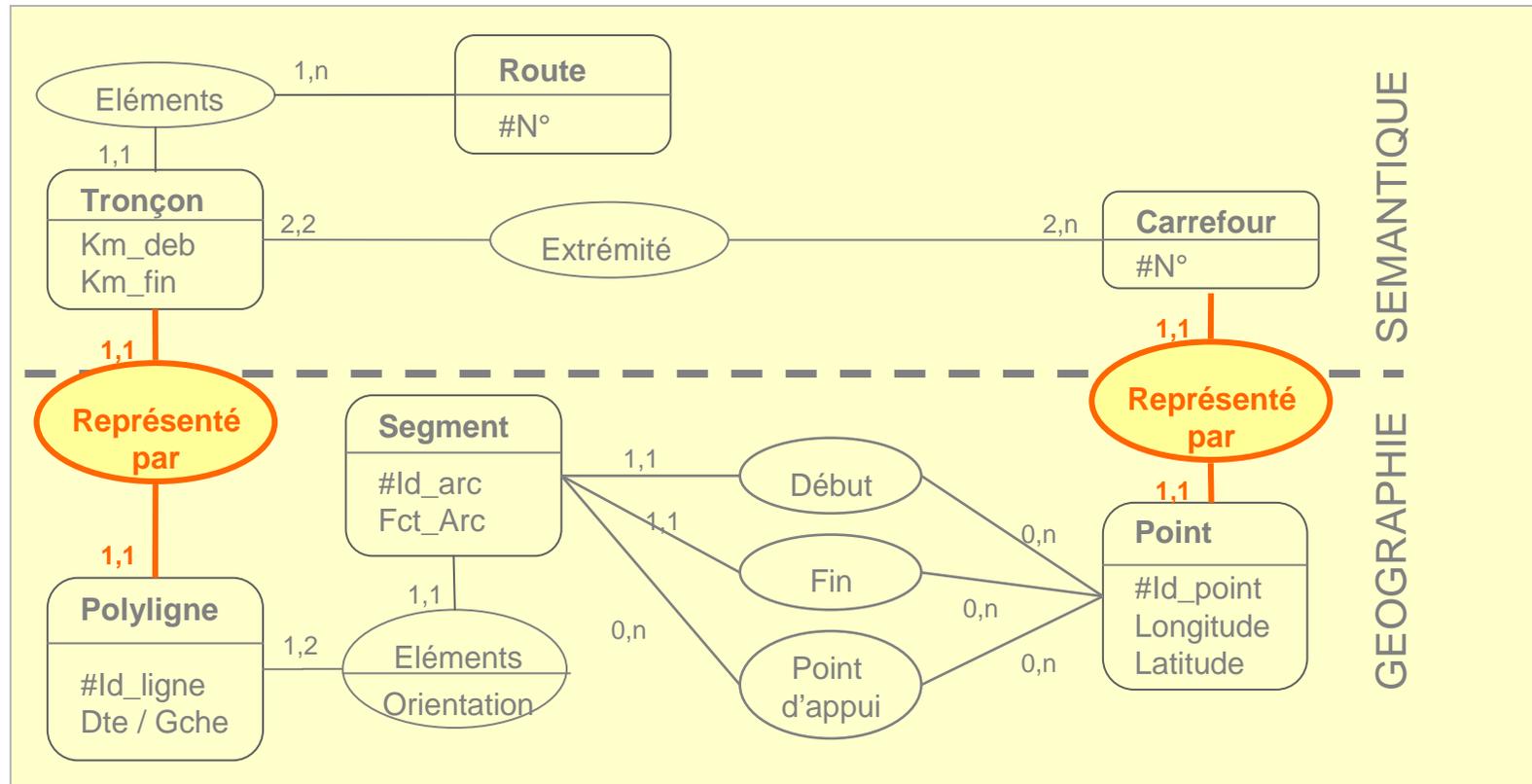
www.info.univ-tours.fr/~antoine

Sommaire

- **SIG : implémentation pour requêtes spatiales optimisées** 4
- **Requêtes en mode vectoriel : cas d'école** 7
- **Requêtes attributaires : SQL** 9
- **Requêtes spatiales : SQL étendu** 11
- **Index spatiaux** 16

SIG : implémentation

Modélisation de l'information géographique (rappel)



En théorie – MCD représentable dans un SGBD relationnel

En pratique – Peu adapté à la représentation de la partie géographique :

- Pas d'optimisation pour l'accès à l'information spatiale
- Requêtes spatiales de trop bas niveau

SIG : implémentation

SIG : couplage SGBD-R / SI spatial

- SGBD-R : information sémantique + lien vers l'information géographique
- SI spatial : information géométrique

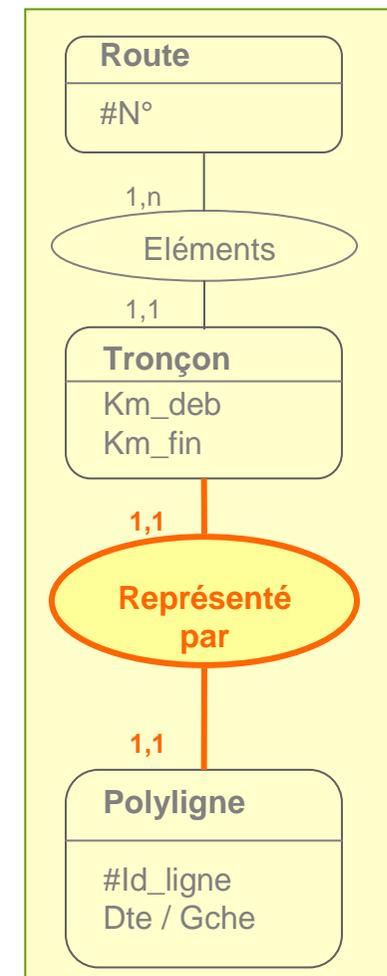
Requêtes dans les SIG : extension de SQL

- types abstraits de données (point, ligne, zone)
- Fonctions d'interrogation spatiales

```
Troncon
{
  num_troncon : integer,
  num_route : integer,
  km deb : integer,
  km fin : integer,
  obj_geom : ligne
}
```

MapInfo
obj

```
CREATE TABLE Troncon ...
```



SIG : implémentation

SQL étendu : exemples de fonctions spatiales

```
PointInZone(zone,point) → booleen  
LineInZone(zone,zone) → booleen  
ZoneInZone(zone,zone) → booleen  
IntersectZone(zone,zone) → booleen  
IntersectLineZone(line,zone) → booleen  
Adjacent(zone,zone) → booleen  
Surface(zone) → real  
Perimeter(zone) → real  
Intersection(zone, zone) → zone  
Centroid(zone) → zone
```



MapInfo Professional®

```
} Within  
} Contains {Entire}  
} Intersects  
Area  
Perimeter  
Centroid
```

```
PointInLine(line,point) → booleen  
IntersectLineZone(line,zone) → booleen  
Length(zone) → real
```

```
Distance(point,point) → real
```

```
Distance
```

Requêtes en mode vectoriel : cas d'école

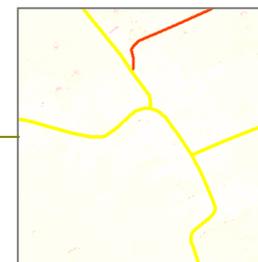
Id_parcelle	Proprietaire	Surface	obj
041_345	Ant_1_66_12	483	*

Cadastre



Id	Long	Type	Trafic	obj
N14	173	Nationale	4 500	*

Voirie



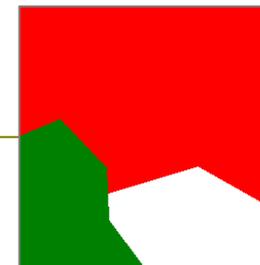
Id	Type	Superficie	obj
041_26	Mairie	483	*

Bâti



Id	Nom	Dep	Pop	obj
41000	Blois	41	51500	*

Communes



Requêtes en mode vectoriel : cas d'école

Exemples de requêtes

1. Quelle est la longueur de la nationale 14 ?
2. Quelle est le trafic moyen des routes départementales ?
3. Quelle est la ville la plus peuplée du Loir-et-Cher ?
4. Quelle est la taille moyenne des villes dans chaque département ?

5. Quelles sont les routes qui croisent la nationale 14 ?
6. Quelles communes disposent encore d'une église ?
7. Quelles sont les communes traversées par la nationale 14 ?
8. Combien y-a-t-il d'écoles dans la ville de Blois ?
9. Combien y-a-t-il de propriétés cadastrales à Blois ?

REQUETES
ATTRIBUTAIRES

REQUETES
SPATIALES

Requêtes attributaires : SQL

Requêtes attributaires unitables

Id	Long	Type	Trafic	obj
N14	173	Nationale	4 500	*

Voirie



Quelle est la longueur de la nationale 14 ?

```
SELECT   Long
FROM     Voirie
WHERE    Id= 'N14';
```

Quel est le trafic moyen des routes départementales ?

```
SELECT   AVG(Trafic)
FROM     Voirie
WHERE    Type = 'Départementale';
```

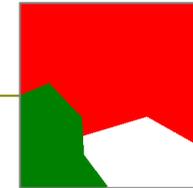
Requêtes attributaires : SQL

Jointures et fonctions d'agrégat

Id	Nom	Dep	Pop	obj
41000	Blois	41	51500	*

Quelle est la ville la plus peuplée du Loir-et-Cher ?

Communes



```
SELECT Nom
FROM Communes
WHERE Pop IN (
    SELECT Max(Pop)
    FROM Communes WHERE Dep = '41')
AND Dep = '41';
```

Quelle est la taille moyenne des villes dans chaque département ?

```
SELECT Dep, AVG(Pop)
FROM Communes
WHERE Pop > 3000
GROUP BY Dep;
```

Requêtes spatiales : SQL étendu

Opérateurs spatiaux

Id	Long	Type	Trafic	obj
N14	173	Nationale	4 500	*

Voirie



Quelles sont les routes qui croisent la nationale 14?

```
SELECT Id
FROM Voirie
WHERE obj Intersects(
    SELECT obj
    FROM Voirie
    WHERE Id = "N14" )
);
```

Requêtes spatiales : SQL étendu

Jointure spatiale

Quelles communes possèdent une église ?

```
SELECT Nom
FROM Communes
WHERE obj Contains(
    SELECT obj
    FROM Batiments
    WHERE Type = "Eglise" )
);
```

```
SELECT Communes.Nom
FROM Communes, Bati
WHERE Communes.obj Contains Bati.obj
AND Bati.Type = "Eglise" ;
```

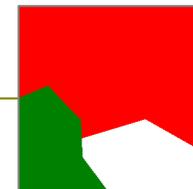
Id	Type	Superficie	obj
041_26	Mairie	483	*

Bati



Id	Nom	Dep	Pop	obj
41000	Blois	41	51500	*

Communes



Quelles sont les communes traversées par la nationale 14 ?

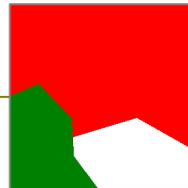
↳ Requête semblable à la précédente : la jointure se fait ici avec INTERSECTS

Requêtes spatiales : SQL étendu

Jointure spatiale et fonction d'agrégats

Id	Nom	Dep	Pop	obj
41000	Blois	41	51500	*

Communes



Id	Type	Superficie	obj
041_26	Mairie	483	*

Bati



Combien y-a-t-il d'écoles dans la ville de Blois ?

```
SELECT Count(Bati.obj)
FROM Communes, Bati
WHERE Communes.obj Contains Bati.obj
AND Bati.Type = "Ecole"
AND Communes.Nom = "Blois" ;
```

Combien y-a-t-il de propriétés cadastrales dans la ville de Blois ?

↳ Requête semblable à la précédente avec les tables Cadastre et Communes

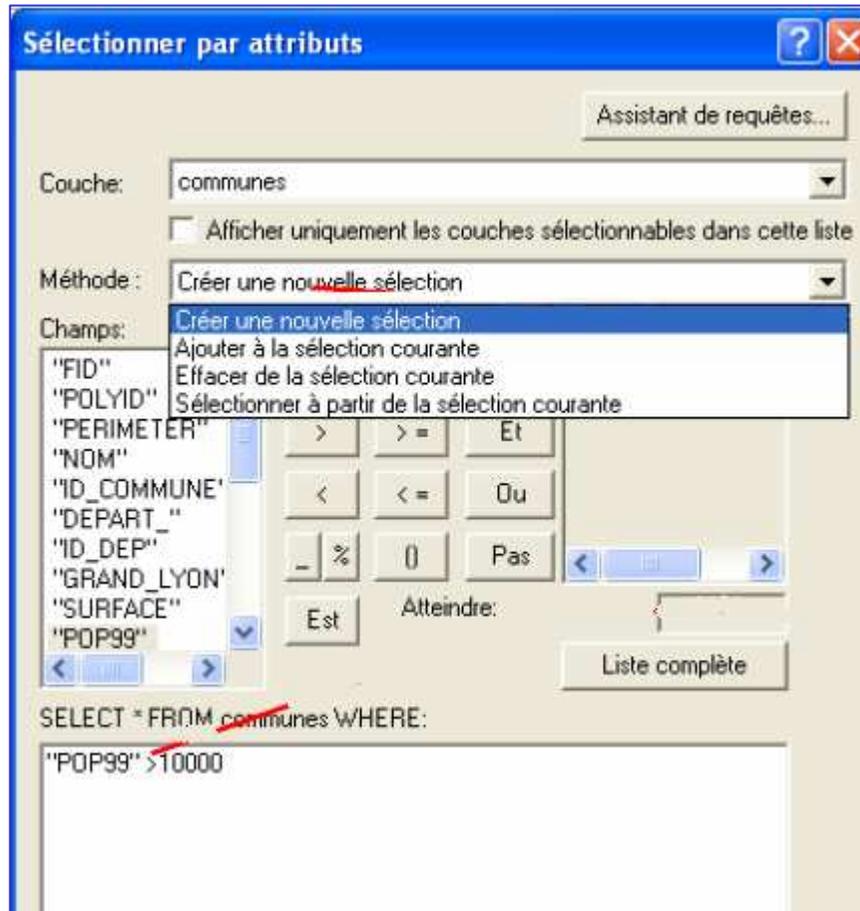
Requêteurs SIG

Requêteur MapInfo Professional® (v. 8.0)

The image shows two overlapping screenshots of the 'Sélection SQL' dialog box in MapInfo Professional v. 8.0. The top screenshot shows the main dialog with fields for 'Colonnes', 'Tables', and 'Critères'. The 'Opérateurs' dropdown menu is open, and a red circle highlights the following items: '()', 'And', 'Or', 'Not', 'Like', 'Contains', 'Contains Entire', 'Within', 'Entirely Within', and 'Intersects'. A red arrow points from the text 'opérateurs spatiaux' to this circle. The bottom screenshot shows the 'Fonctions' dropdown menu open, with a red circle highlighting 'CartesianProduct', 'CartesianPerimeter', 'CentroidX', 'CentroidY', 'Chr\$', 'Cos', 'CurDate', 'Day', 'DeformatNumber\$', and 'Distance'. A red arrow points from the text 'fonctions spatiales' to this circle. Both screenshots include buttons for 'OK', 'Annuler', 'Effacer', 'Vérifier', and 'Aide', as well as 'Sauver Modèle' and 'Charger Modèle' buttons.

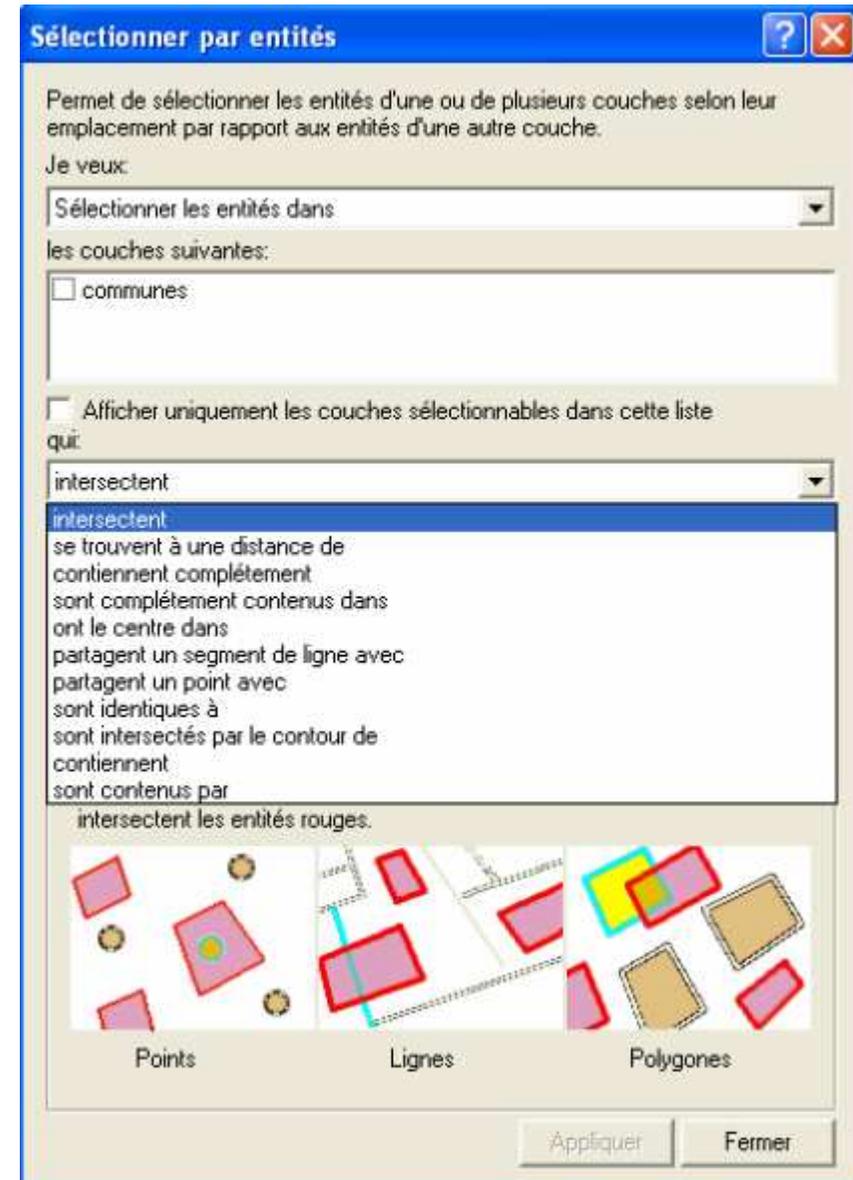
Requêteurs SIG

Requêteur ArcGIS



↑ Requêteur attributaire

Requêteur spatial ⇔



Index spatiaux

Optimisation de requêtes

Problème – Temps d'exécution d'une requête

Exemples

- Liste des villes de plus de 10 000 habitants

⇒ Index classique (SGBD-R) sur l'attribut *Pop*

- Pointé sur une zone dans une couche donnée
- Sélection de zone dans un rectangle donné

Problème : Comparer les coordonnées de la sélection avec celles de tous les objets de la couche : nombreux tests à faire et test unitaire d'appartenance très coûteux en temps de calcul

⇒ **Index spatial**

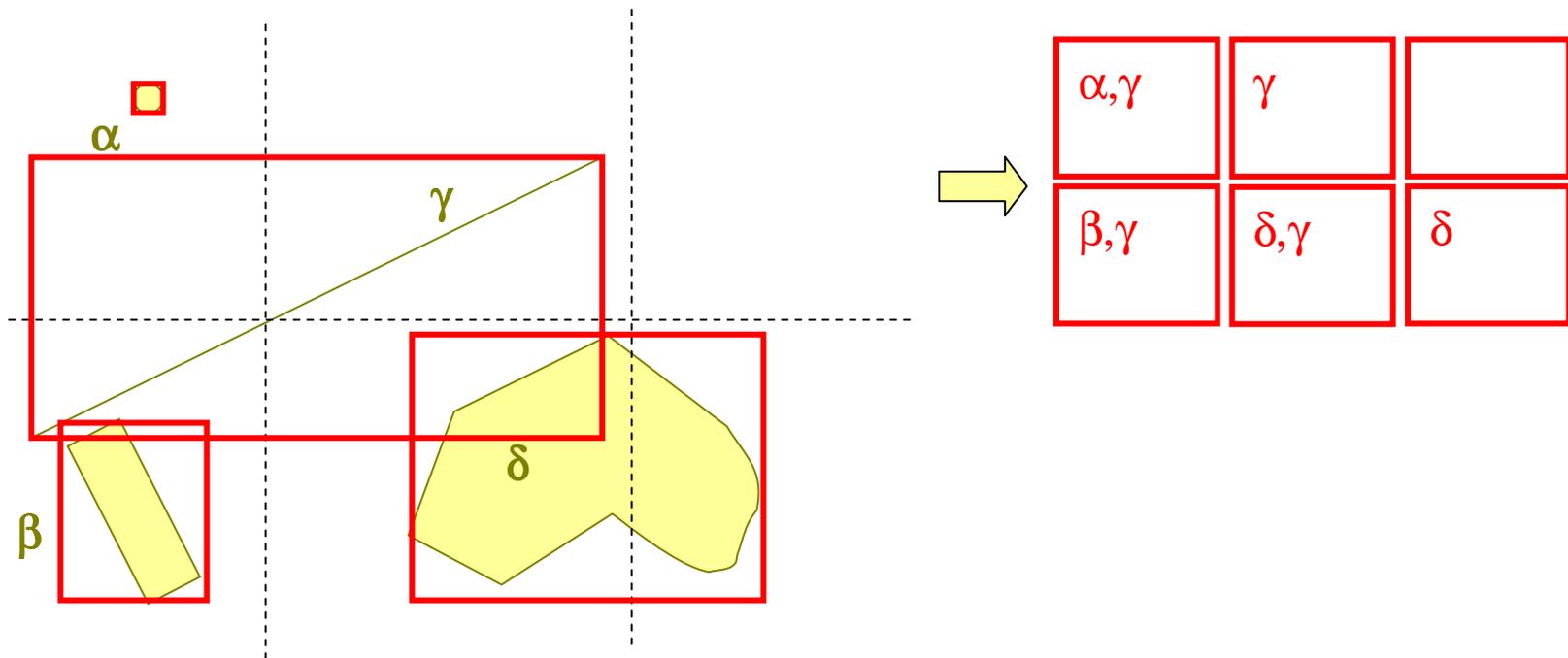
Id	Nom	Dep	Pop	obj
41000	Blois	41	51500	*



Index spatiaux

Principes

Idée – Partitionner l'espace pour limiter la recherche aux objets d'une sous-zone considérée \Rightarrow index spatial défini sur ces zones

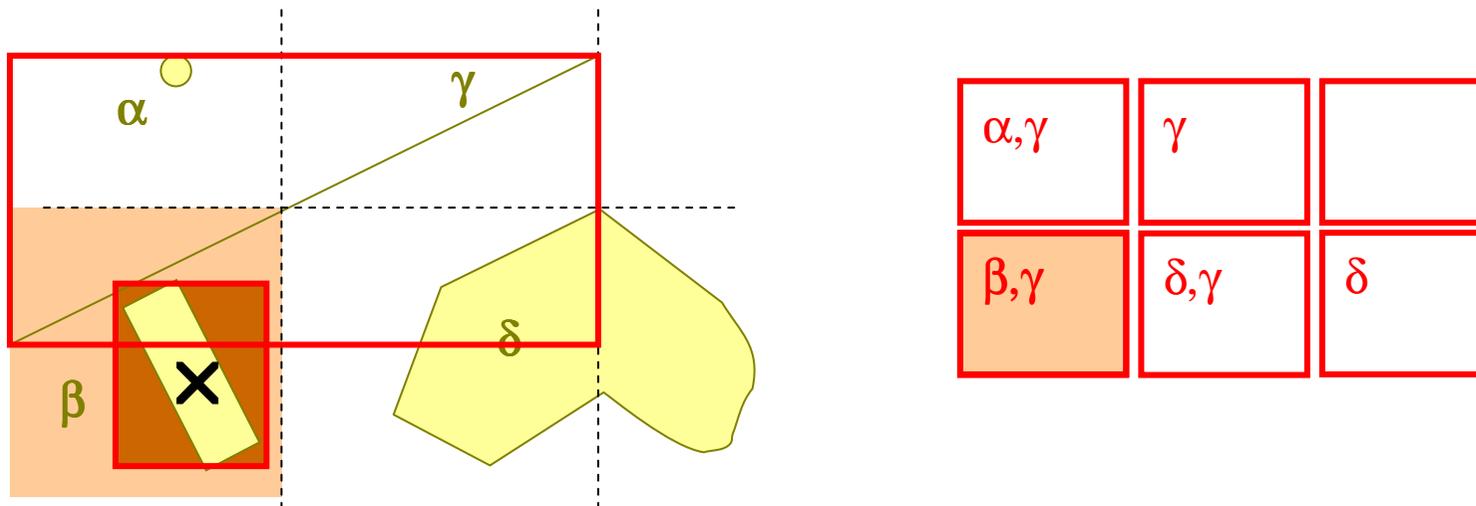


Index spatial – Liste des objets en intersection (inclus, coupant, ou contenant) avec chaque zone. Plus précisément, indexation de leur rectangle englobant minimal (*rem*).

Index spatiaux

Sélection d'objets avec indexation : cas du pointage

1. Détermination de la zone concernée par le pointage
2. L'index donne les *rem* correspondant
3. Recherche des *rem* englobant bien le point désigné
4. Test d'appartenance du point à l'objet du *rem*.



Principales techniques d'indexation spatiale

Seule la constitution des zones d'indexation varient d'une technique à l'autre

- Structures en grille (*grid index*)
- Arbres quaternaires (*Quadtrees*)
- Arbres R (*R trees*)

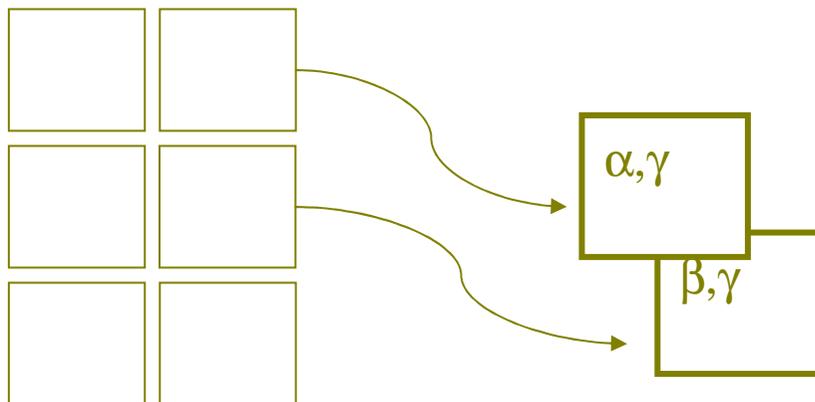
Index spatiaux

Index à grille régulière

[Bentley & Friedman. 1979]

Structure d'indexation la plus simple : on partitionne une fois pour toute l'espace par une grille régulière.

Implémentation – Coordonnées grille + tableau 2D de pointeurs vers des pages mémoires décrivant tous les *rem* de chaque zone de la grille



Ajout d'un objet – Ajout dans les pages des zones concernées. Si page déjà pleine, page(s) de débordement

Pointage – Accès à l'index aisé à partir des coordonnées du point désigné

Efficacité – Puissant si répartition homogène des objets et grille adaptée à la taille des pages. Bien moins performant sinon

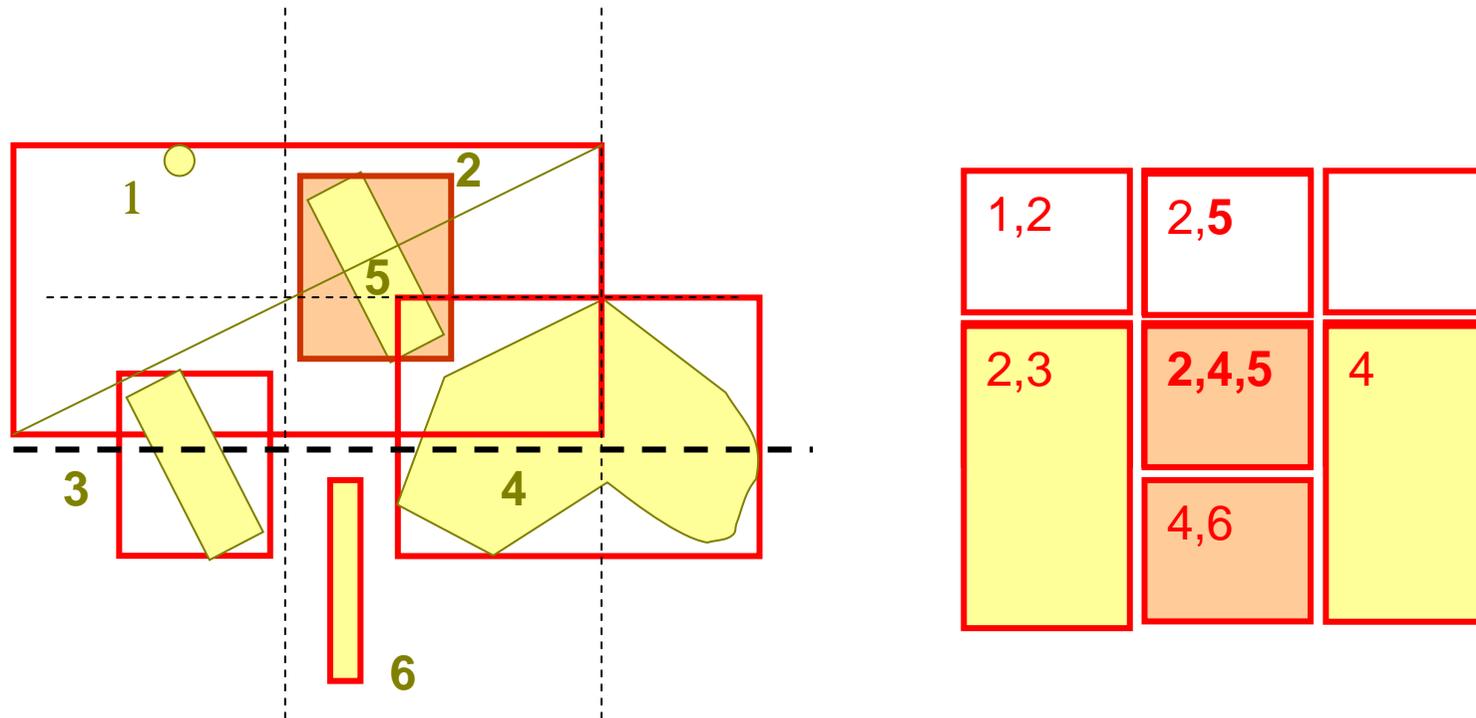
Index spatiaux

Index à grille adaptative

[Nivergelt et. al. 1994]

Idée – Redécoupage dynamique de la grille si trop grande densité d'objets dans une zone donnée

Exemple : index à grille adaptative avec un maximum de 3 éléments par zone



Pointage – Aussi aisé que pour une grille fixe (découpage à instant t connu)

Index spatiaux

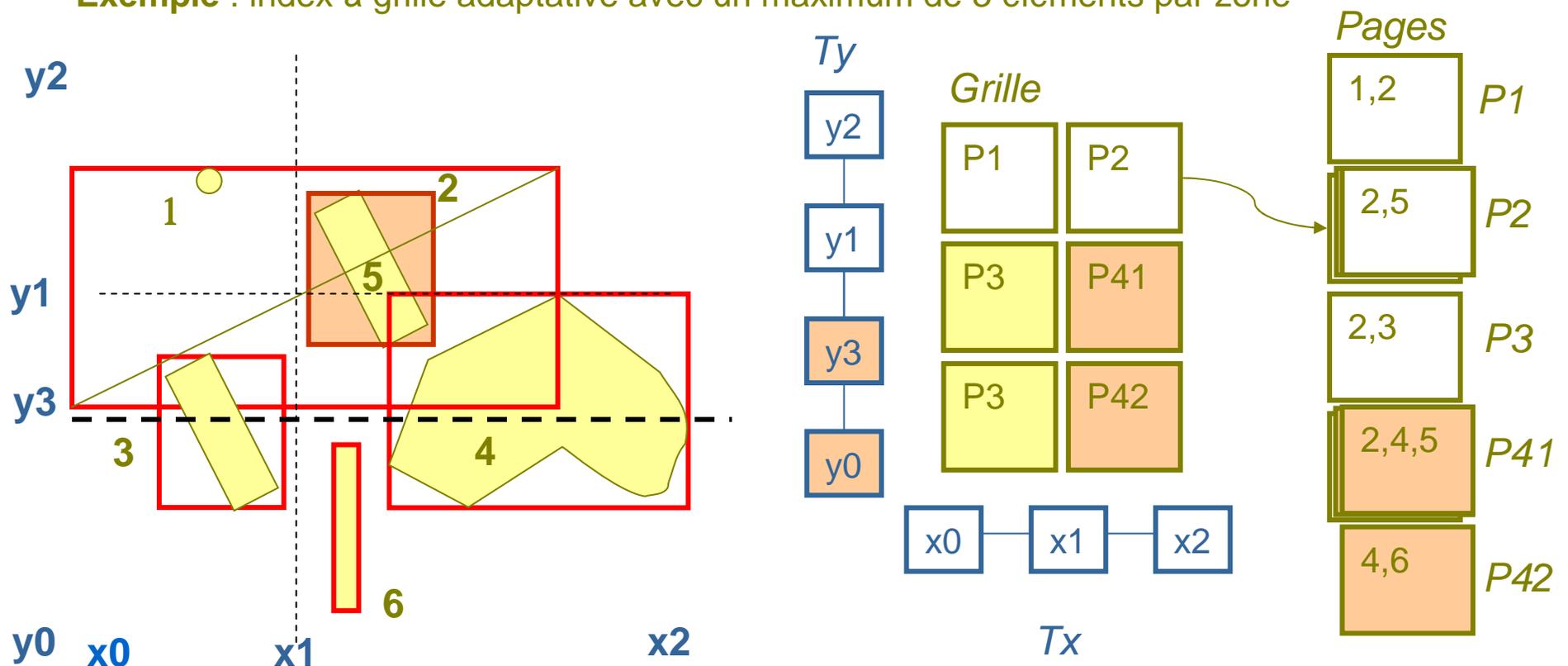
Index à grille adaptative

[Nivergelt et. al. 1994]

Implémentation – Index composé de trois structures de données :

- Tableau *Grille* pointant sur les pages de zone comme pour une grille régulière
- Vecteurs *Tx* et *Ty* donnant les abscisses et ordonnées de découpage.

Exemple : index à grille adaptative avec un maximum de 3 éléments par zone



Index spatiaux

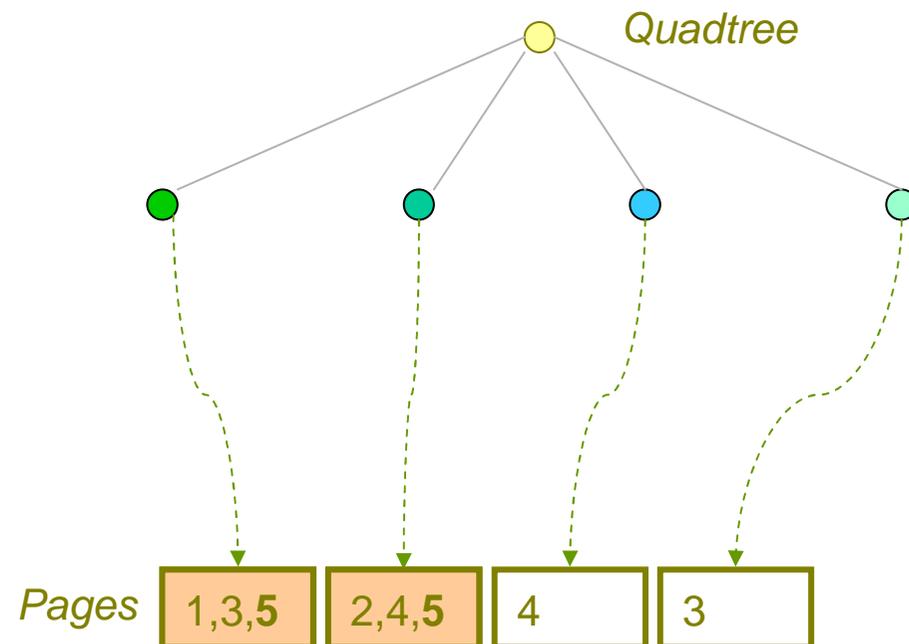
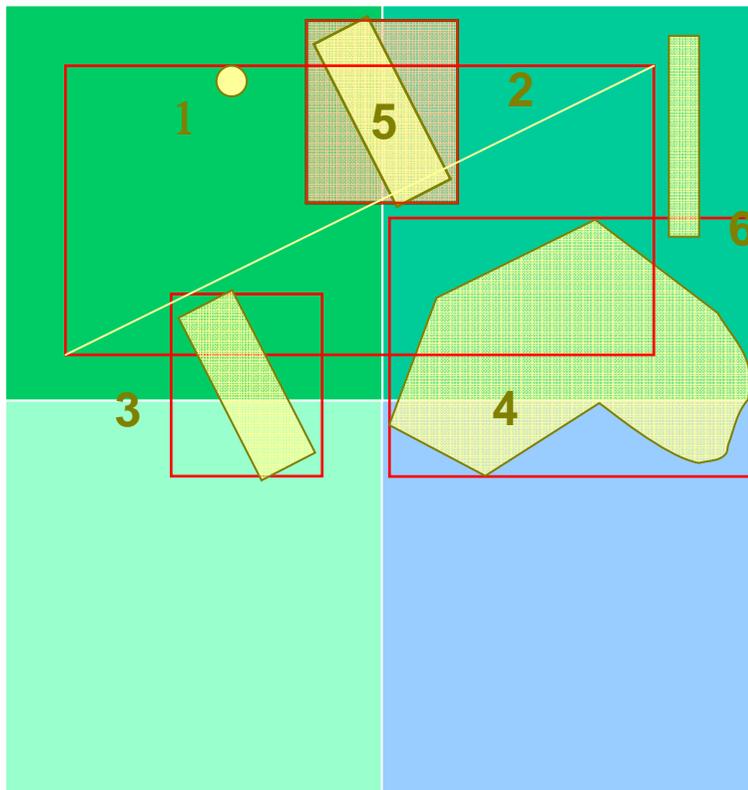
Index à arbre quaternaire (*quadtree*)

[Samet 1990]

Idée – Découpage dynamique : toute zone saturée est redivisée en 4

Implémentation – Découpage décrit par une structure d'arbre

Exemple : index *quadtree* avec un maximum de 3 éléments par zone



Index spatiaux

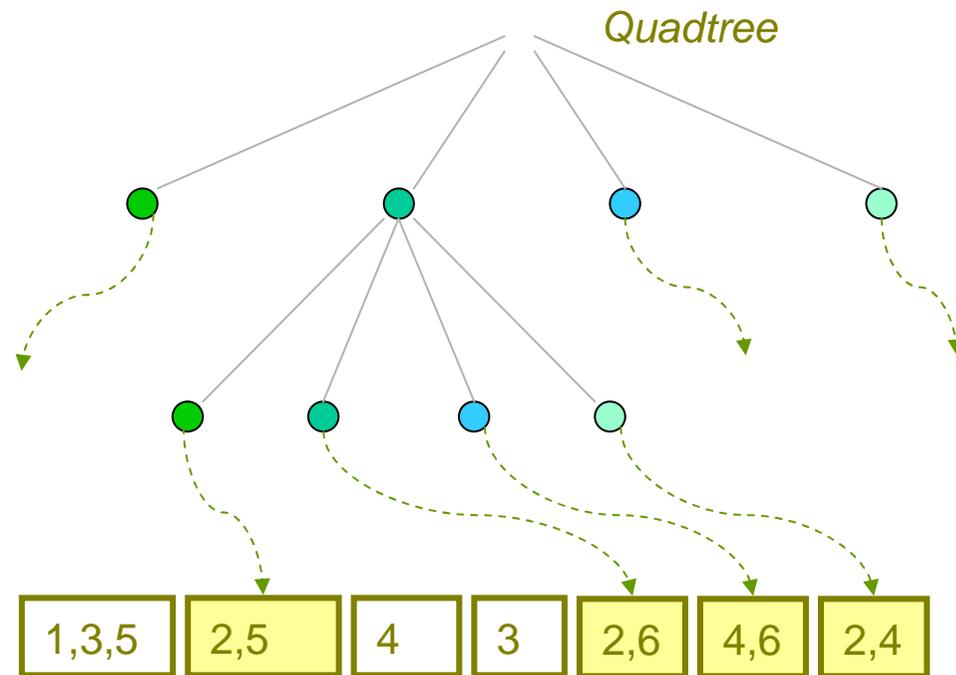
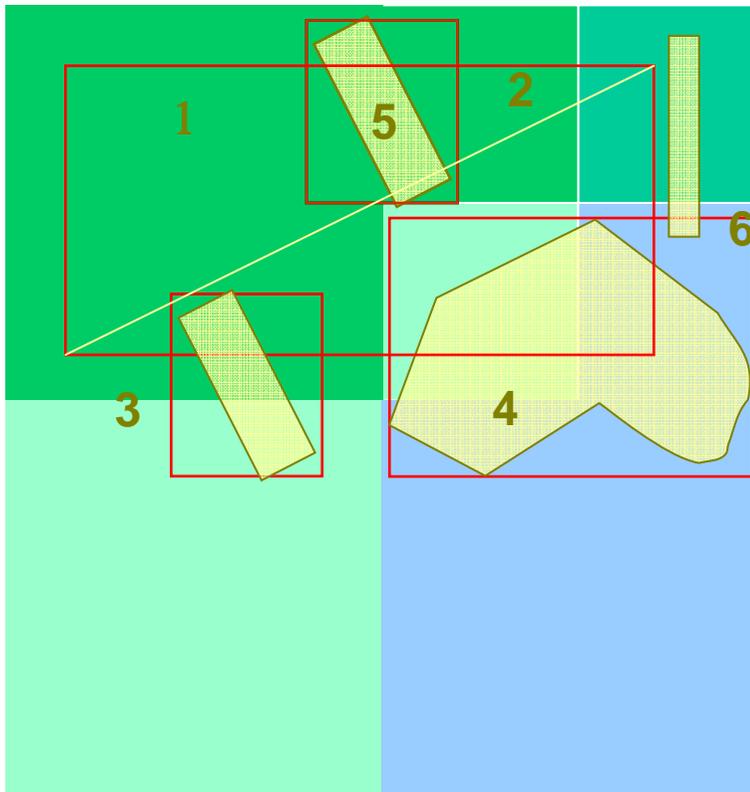
Index à arbre quaternaire (*quadtree*)

[Samet 1990]

Idée – Découpage dynamique : toute zone saturée est redivisée en 4

Implémentation – Découpage décrit par une structure d'arbre

Exemple : index *quadtree* avec un maximum de 3 éléments par zone

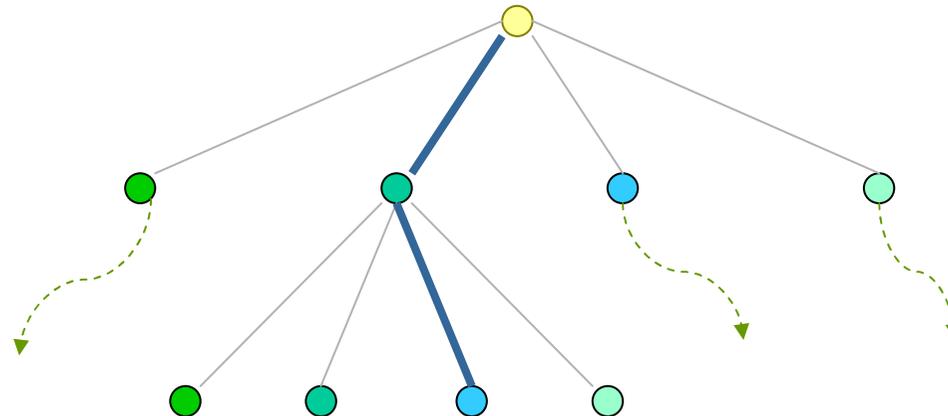


Index spatiaux

Index à arbre quaternaire (quadtree)

[Samet 1990]

Pointage – Détermination de la zone de pointage par parcours de l'arbre (un seul chemin jusqu'à la bonne feuille)



Intérêt et limite – Assez bien adapté aux distributions non homogènes d'objets. Peu cependant conduire à des arbres déséquilibrés.

↳ **Arbre équilibré** : index R

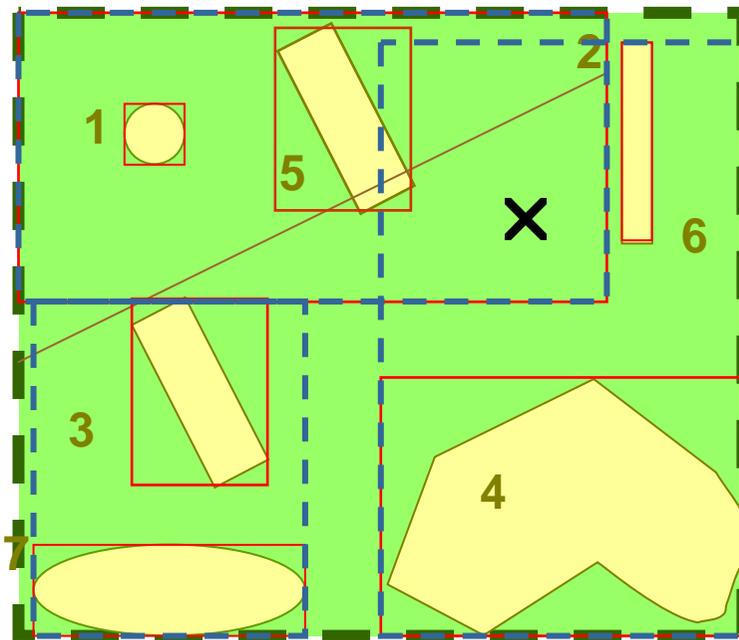
Index spatiaux

Index à arbre R (*R tree*)

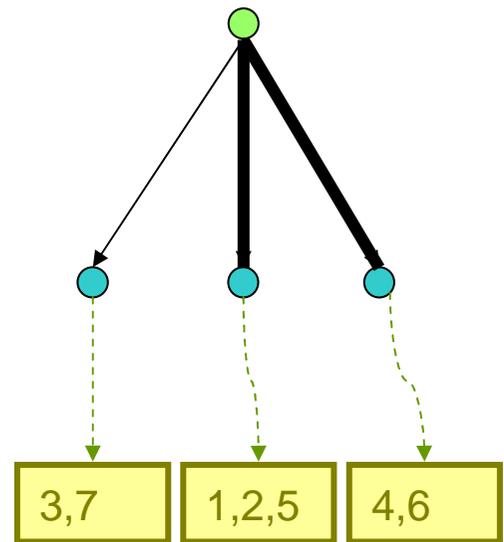
[Guttman 1984]

Principes – Index dynamique avec arbre équilibré

1. Partition incomplète de l'espace : simples zones englobant tous les objets
2. Organisation hiérarchique des zones suivant un arbre équilibré



Chaque zone est le rem des éléments (rem ou objet) englobées



Sélection (pointé) – Plusieurs parcours à étudier dans l'arbre R

Index spatiaux

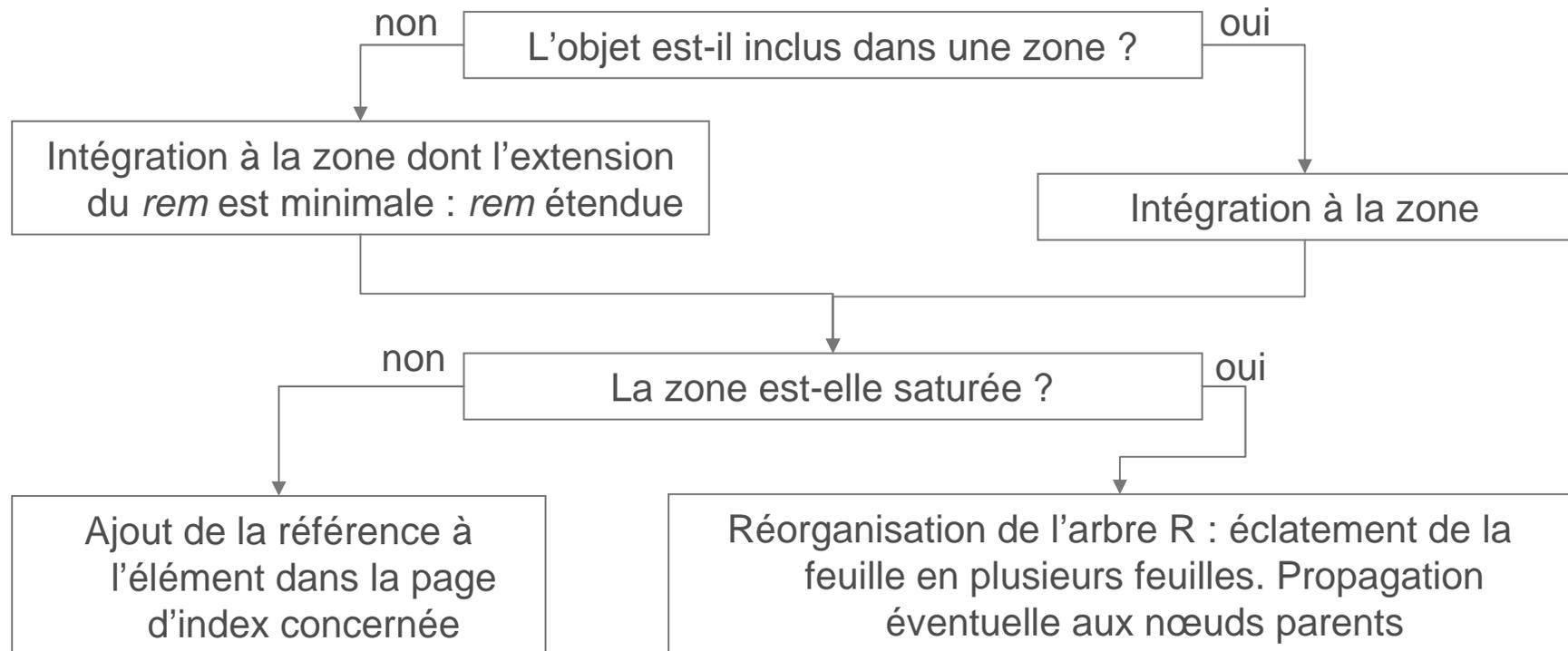
Index à arbre R (*R tree*)

[Guttman 1984]

Arbre R – Arbre équilibré de degré n fixé avec :

- Le nœud racine possède entre 2 et n fils
- Tout nœud interne a entre m et n fils, avec m entier inférieur à $n/2$.

Construction dynamique de l'arbre R – Ajout d'un nouvel élément à l'index



⇒ analogue arbre B : cf. cours BD Avancées

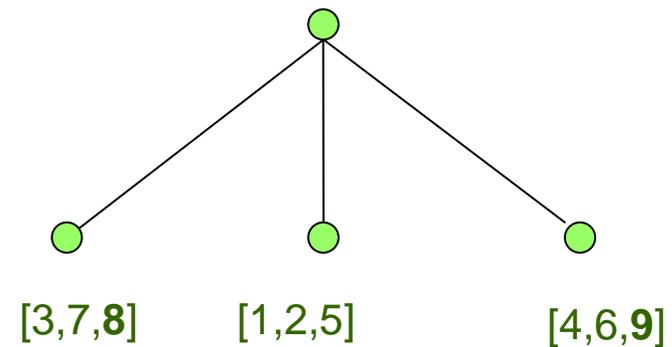
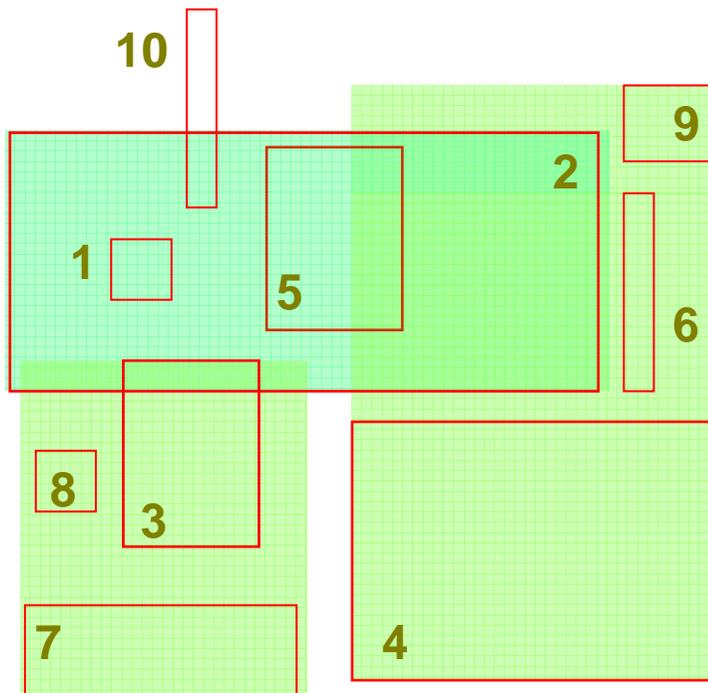
Index spatiaux

Index à arbre R (*R tree*)

[Guttman 1984]

Construction dynamique de l'arbre R – Ajout d'un nouvel élément à l'index

Exemple : arbre R de degré 3

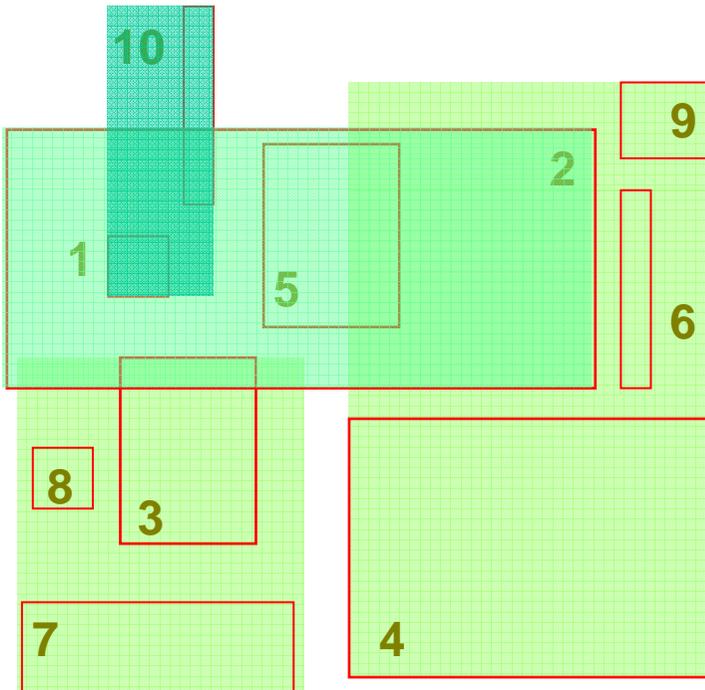


Index spatiaux

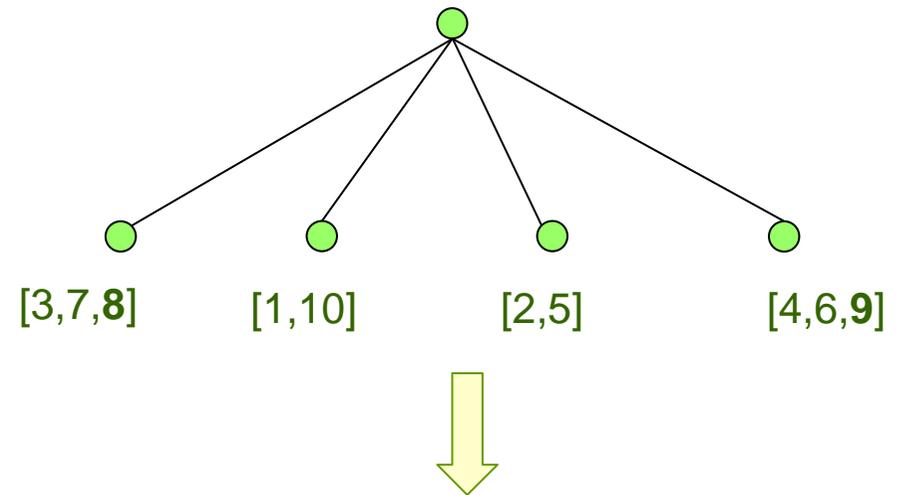
Index à arbre R (*R tree*)

Construction dynamique de l'arbre R

Exemple : arbre R de degré 3



[Guttman 1984]

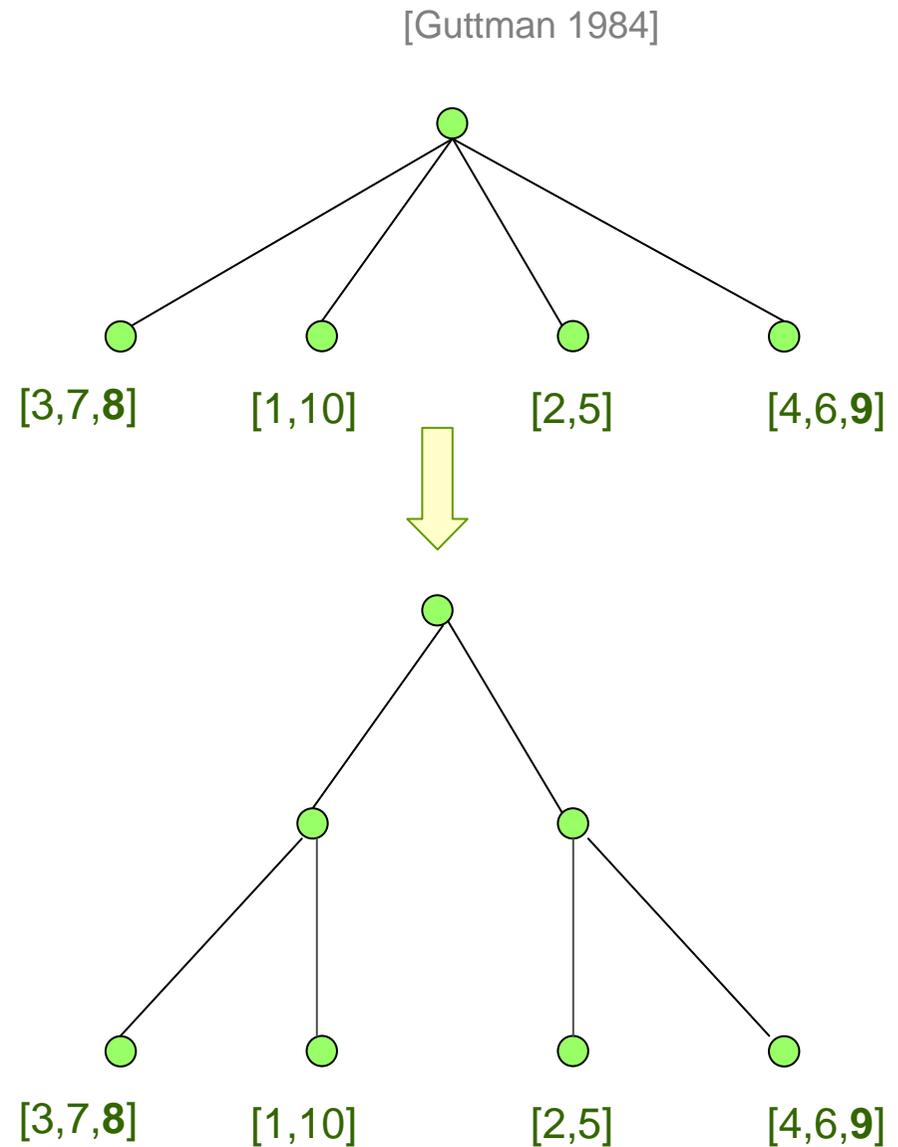
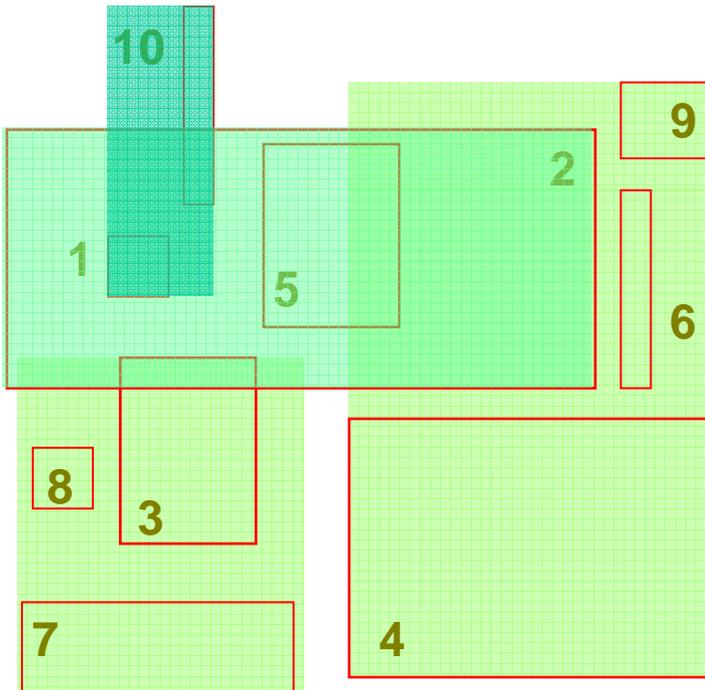


Index spatiaux

Index à arbre R (*R tree*)

Construction dynamique de l'arbre R

Exemple : arbre R de degré 3



Index spatiaux

Index à arbre R (*R tree*)

[Guttman 1984]

Construction dynamique de l'arbre R – Suppression d'un élément à l'index

Idem arbre B : fusion éventuelle si une zone a moins de m éléments.

Construction statique de l'arbre R

Lors de la création d'une BD, il est préférable de construire l'index avec une vision globale plutôt que de s'en remettre à un ajout des objets un à un.

Stratégies envisageables :

- Minimiser le nombre des recouvrements de zone à un niveau de l'arbre, pour limiter les parcours multiples ensuite,
- Minimiser la surface totale des zones, pour éviter de couvrir des zones sans objets
- Maximiser le nombre de fils à chaque nœud, afin de limiter la profondeur du graphe et accélérer son parcours.
- ...

Index spatiaux

Comparaison des différents index

	Grille statique	Grille dynamique	Arbre quaternaire	Arbre R
Complexité d'implantation	★	★	★	
Tolérance à la distribution des objets		😊	😊	★
Performance sélection	★	★	★	★
Situation d'utilisation	Distributions homogènes		Compromis simplicité / efficacité	Performances en toutes situations

Bibliographie

Ouvrages de référence

- **Scholl M., Voisard A., Peloux J.P., Raynal L., Rigaux Ph.** (1996) *SGBD Géographiques : spécificité*. Int. Thomson Publ., Paris, France. Chap. 4

Travaux cités

- **Bentley J.L., Friedman J.H.** (1979) Data structures for range searching. *ACM Computing Surveys*. 11(4).
- **Guttman A.** (1984) R-trees : a dynamic index structure for spatial searching. Proc. *ACM SIGMOD Int. Symposium on the Management of Data*. Pp. 45-67.
- **Nievergelt J., Hinterger H., Sevcik K.C.** (1994) The grid file : an adaptable symmetric multikey file structure. *ACM Trans. on Databases Systems*. 9(1). pp. 38-71.
- **Samet H.** (1990) *Application of spatial data structure*. Addison-Wesley Publ. 9(1).