

KGM/PDB Prostorové spojení

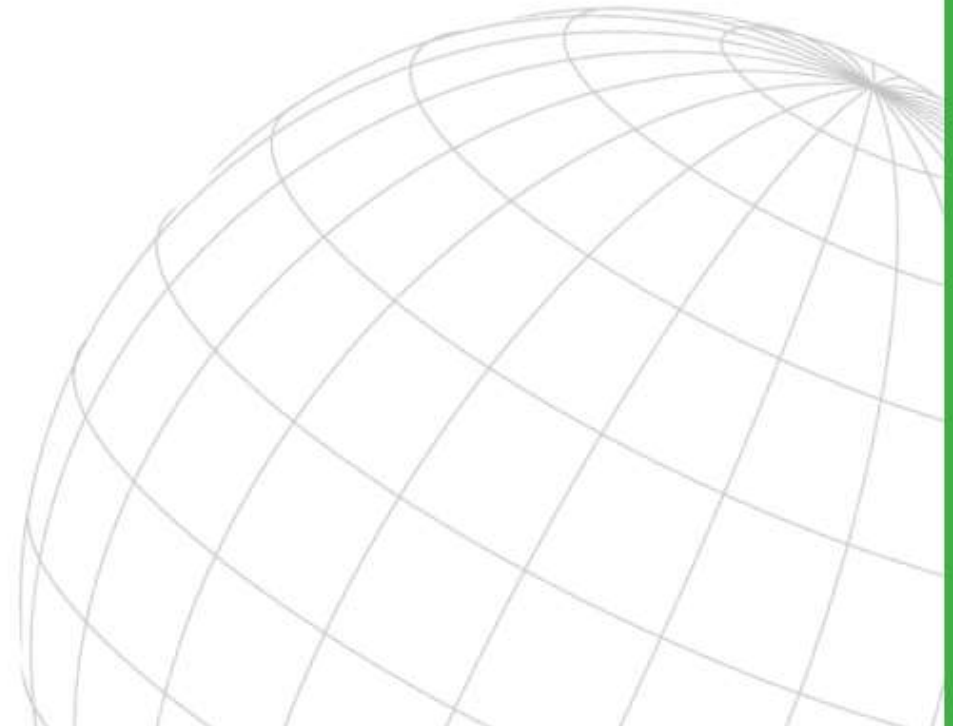
Karel Janečka

Tvorba materiálů byla podpořena z prostředků projektu FRVŠ č. F0584/2011/F1d



Obsah

- Prostorové spojení pomocí hnížděných cyklů
- Prostorové spojení pomocí prostorových indexů



Zpracování prostorových spojení

- Prostorová spojení mají pro prostorové databáze podobnou důležitost, jako operace spojení v klasických databázích.
- Uvažujme tabulku MESTO (nazev, **geometrie**) a tabulku PARKY (nazev, **geometrie**).
- Pak má smysl dotaz: *Najdi všechny parky v daném městě.*

```
SELECT p.nazev
FROM mesta m, parky.p
WHERE m.nazev = 'Plzeň'
AND SDO_RELATE (m.geometrie,p.geometrie,'mask=anyinteract')='TRUE';
```

Zpracování prostorových spojení

- **Prostorový predikát P** , kterým se daný dotaz řeší, je **průnik dvou množin**.
- Platí-li $P(a,b) = \text{TRUE}$, nazveme dvojici objektů (a,b) **hitem**.



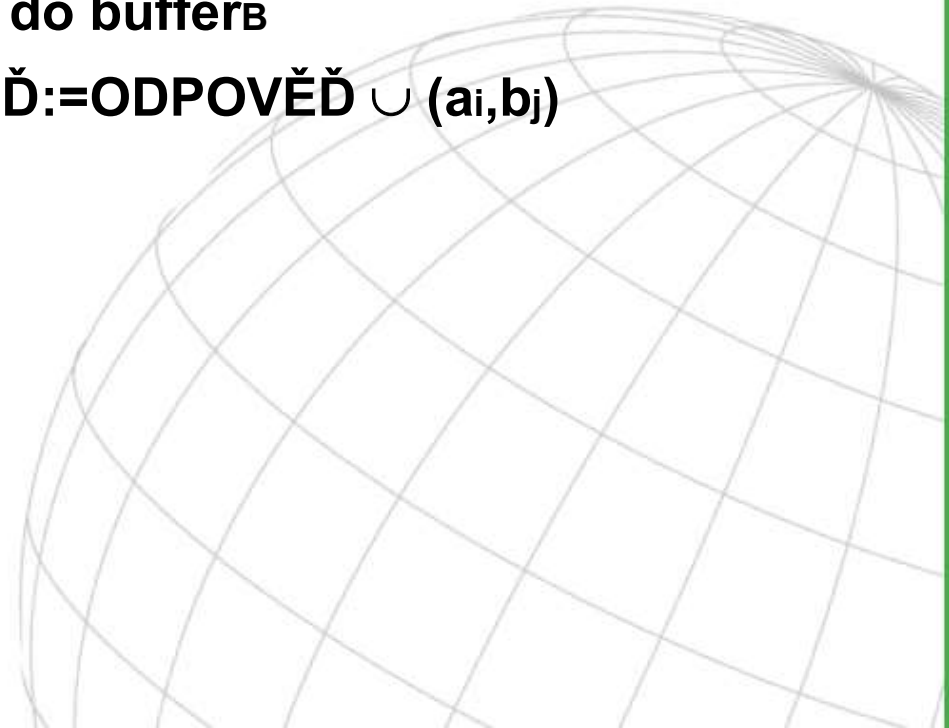
Prostorové spojení pomocí hnízděných cyklů

- Prostorové spojení pomocí hnízděných cyklů je odrazovým můstkem pro další úvahy.
- Vstup: Tabulky A, B s prostorovým atributem.
- Výstup: množina dvojic prostorových objektů, které se překrývají.



Prostorové spojení pomocí hnízděných cyklů

- **begin ODPOVĚĎ:= \emptyset ;**
for i:=1 to m do
 begin přečti objekt a_i z A do buffera
 for j:=1 to n do
 begin přečti objekt b_j z B do bufferb
 if $P(a_i, b_j)$ then ODPOVĚĎ:=ODPOVĚĎ \cup (a_i, b_j)
 end
 end
 end
end;



Prostorové spojení pomocí hnízděných cyklů

- Nevýhodou uvedeného algoritmu může být jeho náročnost časová i paměťová při přenášení objektů do paměti.
- Kvadratická složitost!
- Zaměříme se proto postupně na zpracování spojení, které vlastní přenos objektů do paměti oddaluje a používá ho, jen když už neexistuje jiná možnost.
- Budeme předpokládat objekty organizované podle vhodné indexové struktury, např. R-stromu.

Prostorové spojení pomocí prostorových indexů

- Důležitým aspektem implementace prostorových objektů je jejich reprezentace pomocí aproximací, které slouží jako první krok pro výpočet prostorového spojení.
- Jednou z možností aproximace jsou minimální ohraničující obdélníky (MOO).
- Tyto MOO jsou organizovány ve vhodné datové struktuře – prostorovém indexu (např. R-strom)
- Skutečný vztah dvou objektů **je aproximován vztahem jejich MOO**. Takové spojení nazveme **MOO-spojení**.
- MOO-spojení vrací dvojice objektů, pro které platí:

$$\text{MOO}(a) \cap \text{MOO}(b) \neq \emptyset$$

MOO-spojení

- Algoritmus **MOO-spojení** lze popsat následovně:
- Vstup: R-stromy s kořeny T , resp. U . předpokladem bude stejná hloubka obou R-stromů (z důvodu jednoduššího vyjádření algoritmu).
- Výstup: množina dvojic identifikátorů objektů, jejichž MOO se překrývají.

algoritmus SPATIALJOIN(T, U)

FOR každý prvek $P_T \in$ uzel T **DO**

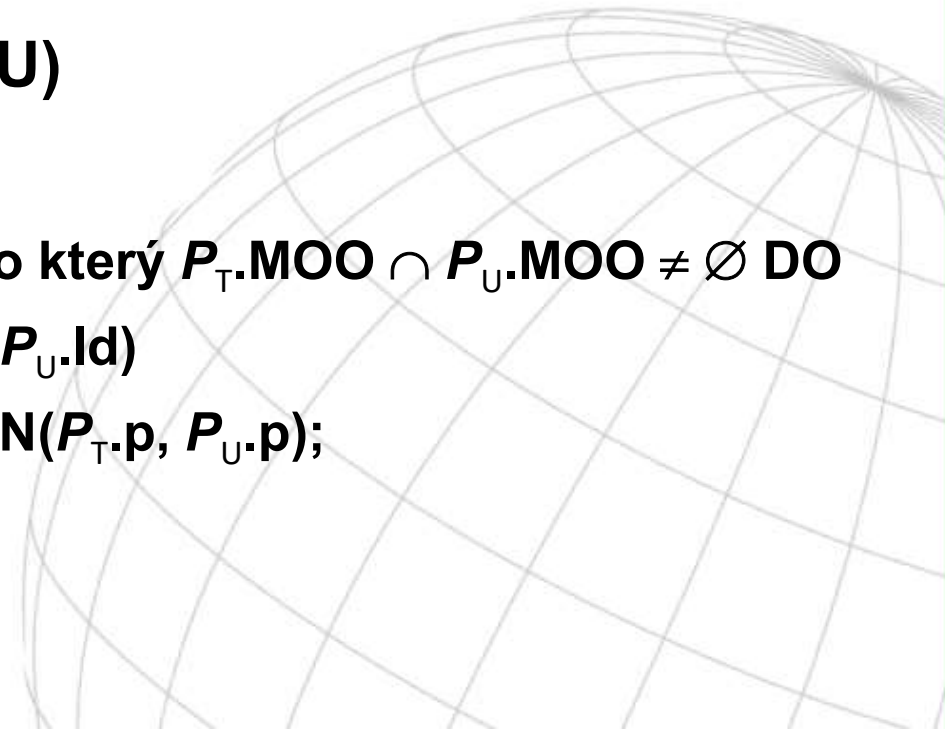
FOR každý prvek $P_U \in$ uzel U pro který $P_T.MOO \cap P_U.MOO \neq \emptyset$ **DO**

IF U je list **THEN** PRINT($P_T.Id, P_U.Id$)

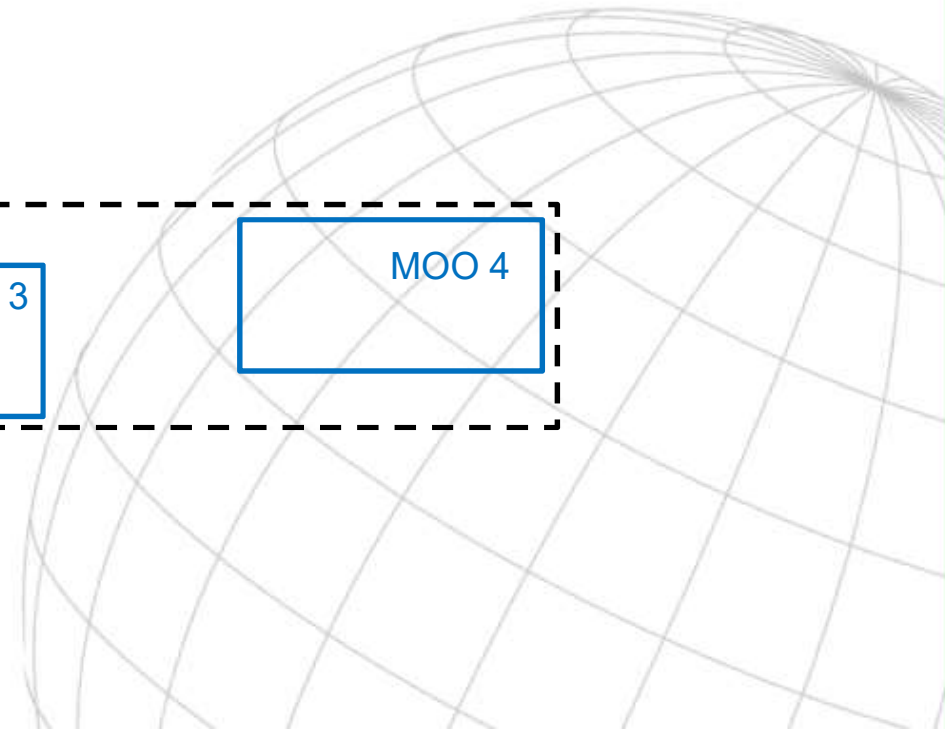
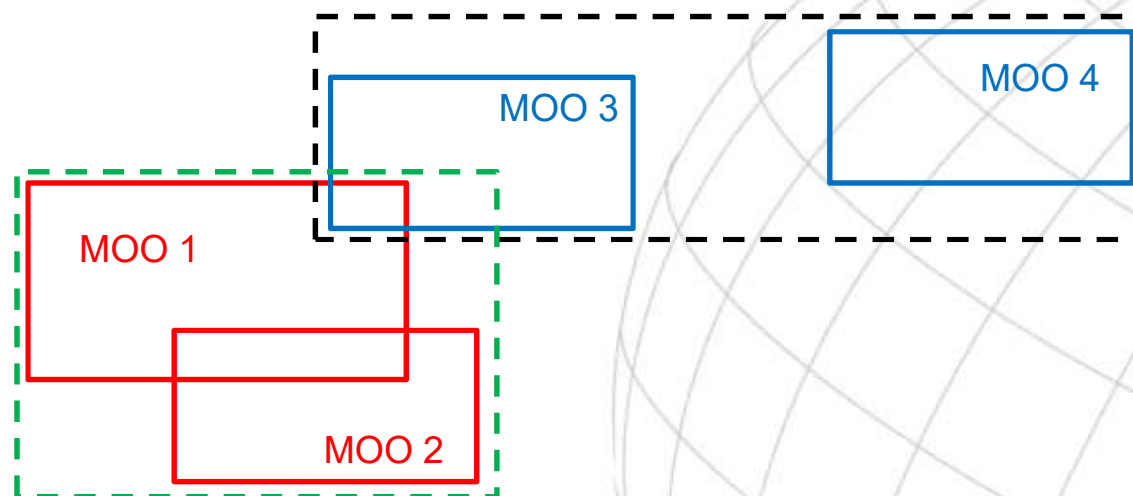
ELSE SPATIALJOIN($P_T.p, P_U.p$);

END

END;

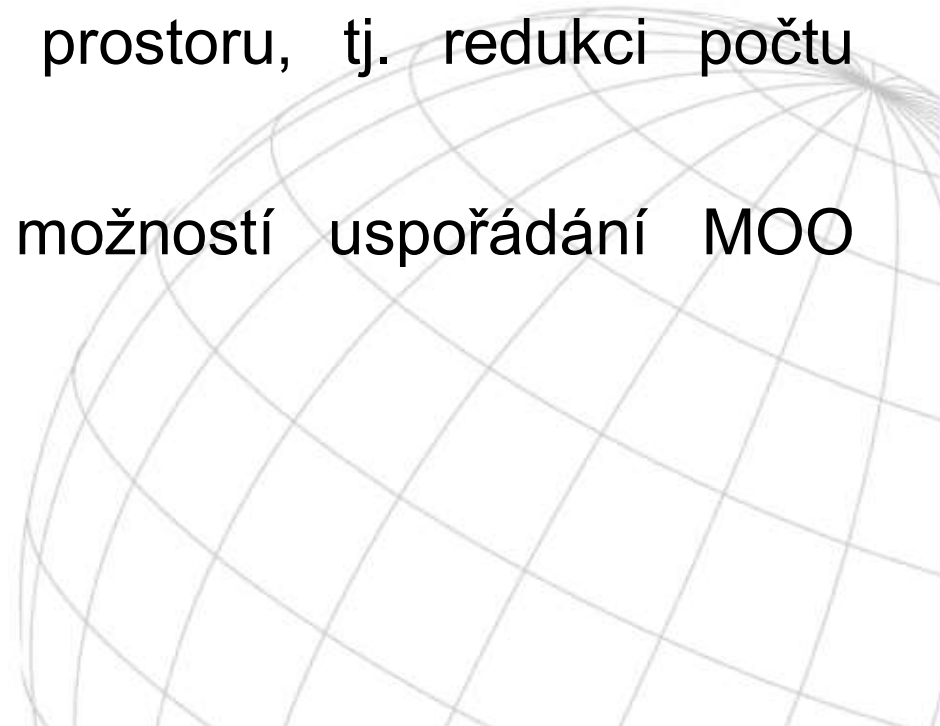


MOO-spojení



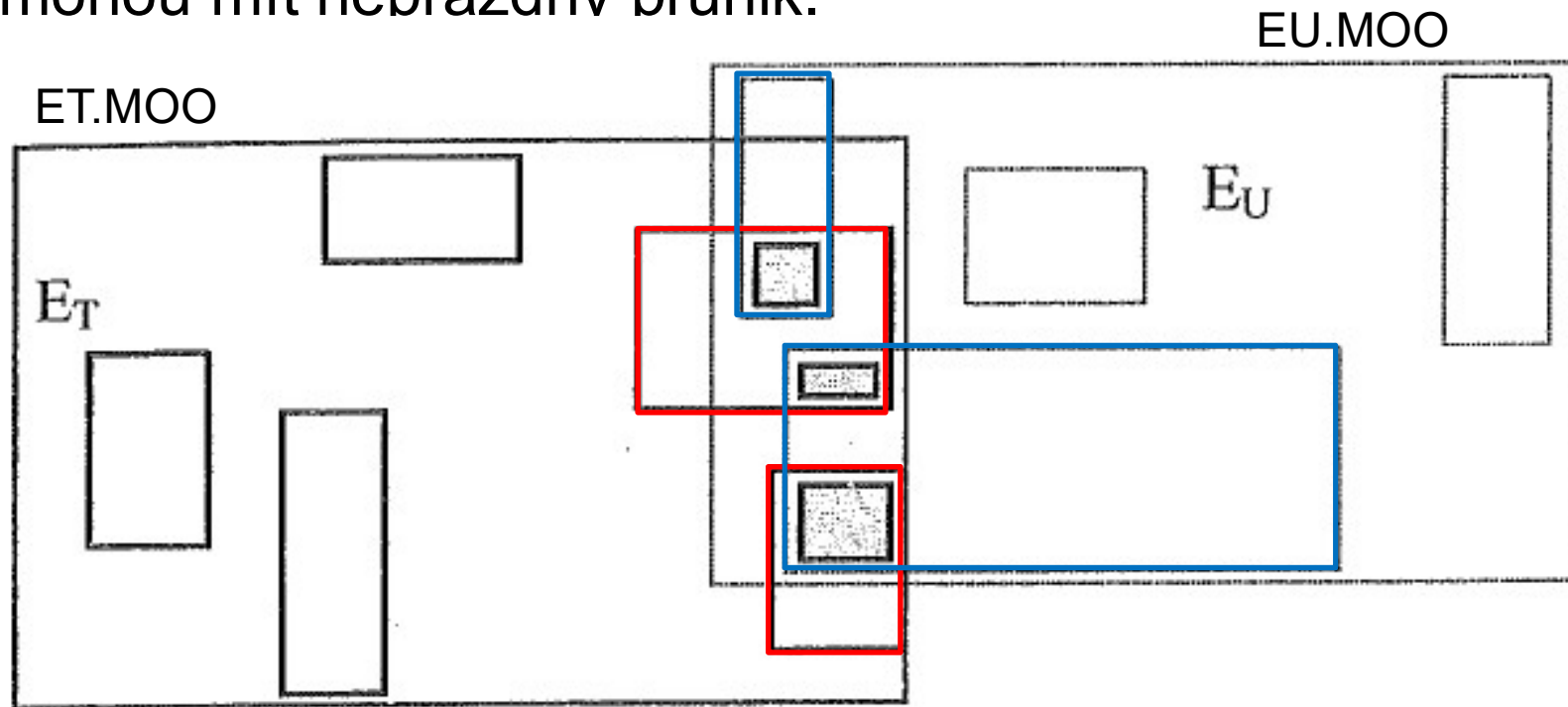
Optimalizace MOO-spojení

- Pro algoritmus MOO-spojení je důležité minimalizovat nejenom počet přístupů na disk, ale i čas CPU, který může při vyhodnocování operace *průnik* vzrůstat.
- Zlepšení algoritmu MOO-spojení bude zahrnovat omezení prohledávaného prostoru, tj. redukci počtu porovnávání dvojic MOO.
- Zlepšení využije jistých možností uspořádání MOO v uzlu.



Optimalizace MOO-spojení 1

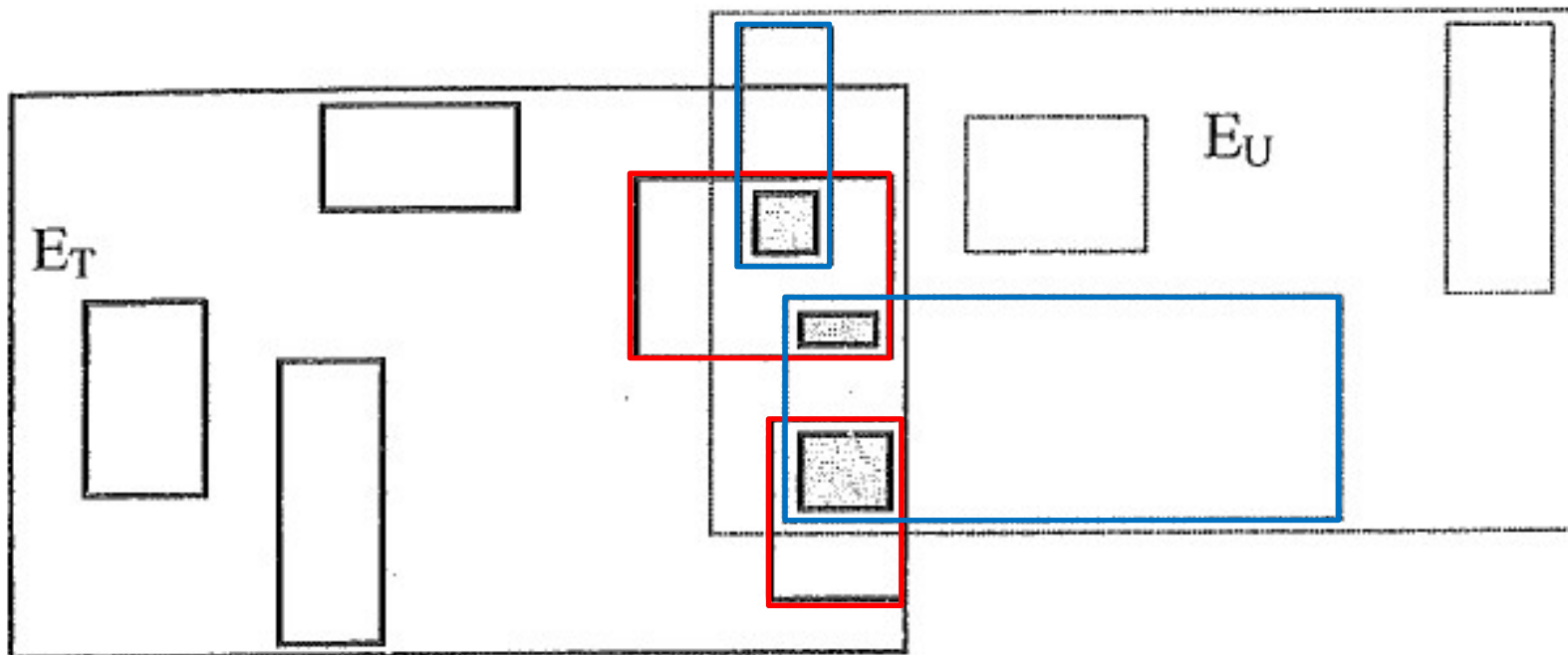
- Tvrzení: Pouze ty dvojice MOO z ET.p a EU.p, z nichž každá má neprázdný průnik s $ET.MOO \cap EU.MOO$, mohou mít neprázdný průnik.



Adepti na neprázdný průnik.

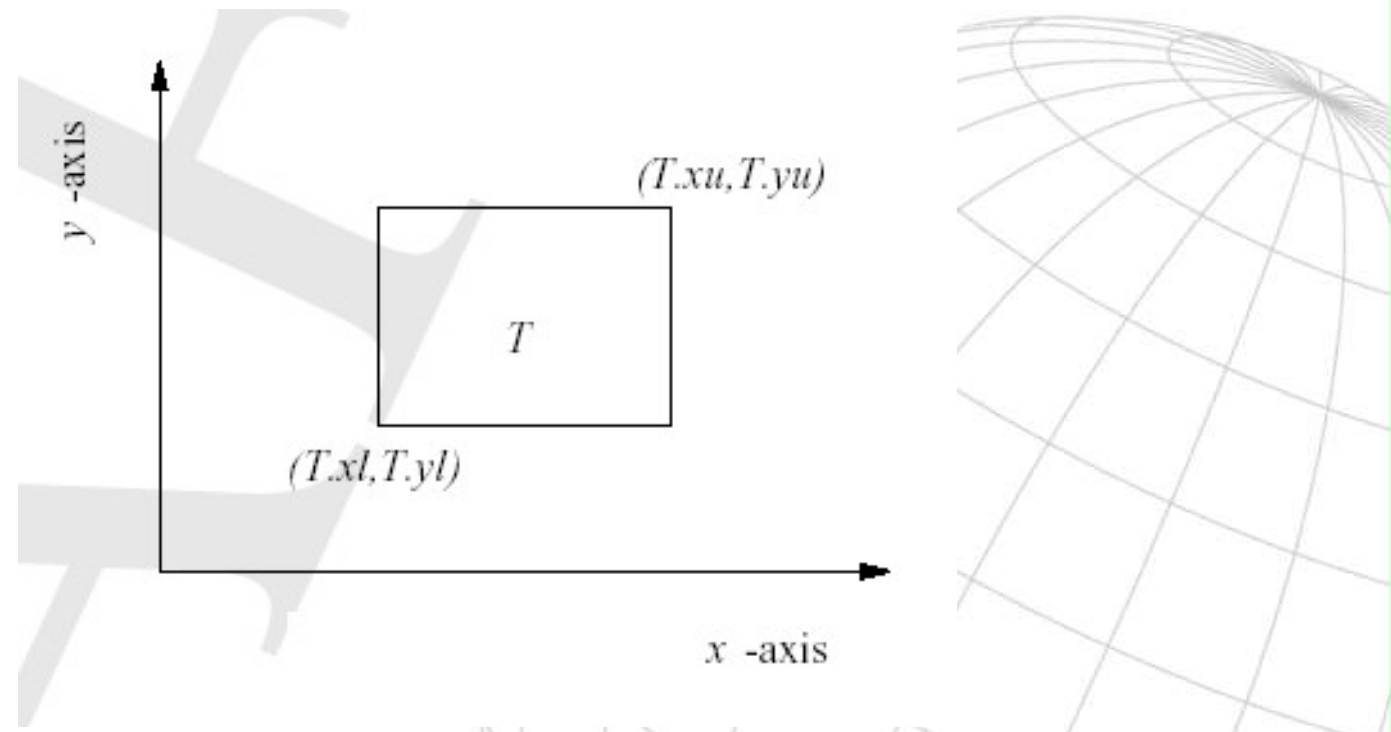
Optimalizace MOO-spojení 1

- Při sekvenčním prohledání každého ze dvou uzlů lze označit ty MOO, které splňují předcházející tvrzení.
- Pak se prostorově porovnávají dvojice, které vzniknou ze všech označených uzlů.



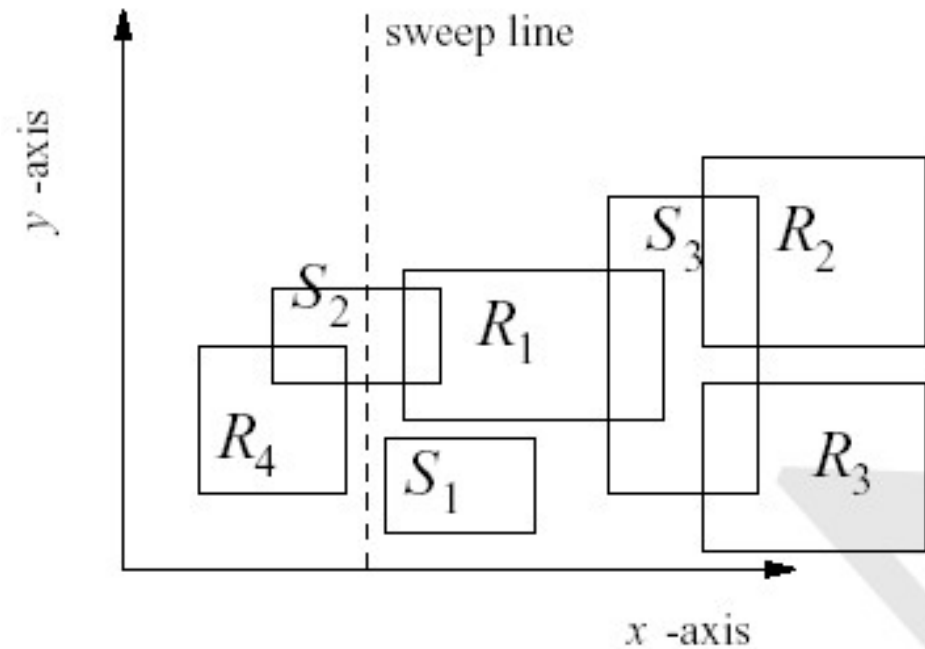
Optimalizace MOO-spojení 2

- Další optimalizace je založena na technice **sweep-line**
- Každý MOO je reprezentován svým dolním levým ($T.xl$, $T.yl$) a pravým horním ($T.xu$, $T.yu$) rohem, vizte obrázek.



Optimalizace MOO-spojení 2

- Vstupní množina MOO (*MOO R1 až R4 a S1 až S3 leží v průniku nadřazených obálek*):



Seřazená množina MOO:

R_4	S_2	S_1	R_1	S_3	R_2	R_3
-------	-------	-------	-------	-------	-------	-------

Optimalizace MOO-spojení 2

- Algoritmus probíhá v následujících pěti krocích:
 - 1) Pohybujeme sweep line (která je kolmá na osu x) směrem zleva doprava a zastavíme se v každém event point ($T_i.x_L$). Tento bod při prvním zastavení sweep-line odpovídá nejmenší hodnotě $T.x_L$ (R_4).
 - 2) Prohledáváme mezi seřazenými MOO z S , dokud nenarazíme na první MOO S_f takový, že $S_f.x_L > T.x_U$. To znamená, že je splněna relace $[T.x_L, T.x_U] \cap [S_j.x_L, S_j.x_U]$ pro všechna $1 \leq j < f$. V našem případě je S_f rovno S_1 . Proto MOO S_2 je kandidát na průnik s R_4 a bude zpracován v dalším kroku.

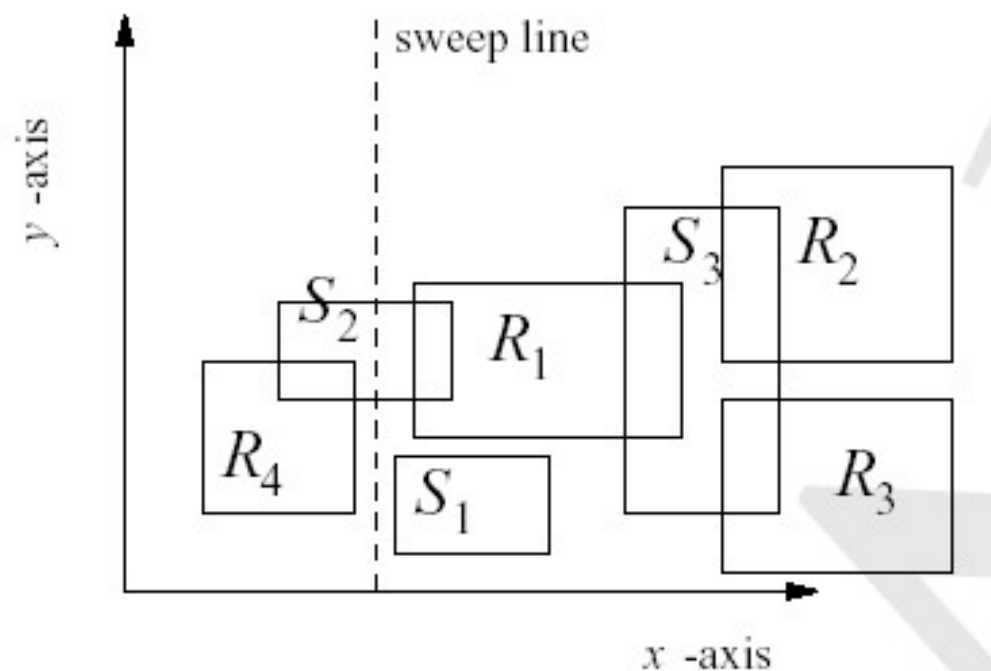
Optimalizace MOO-spojení 2

- 3) Pokud je splněna relace $[T.y_L, T.y_U] \cap [S_j.y_L, S_j.y_U]$ pro všechna j , $1 \leq j < f$, potom MOO S_j má průnik s T . Z toho plyne, že R_4 a S_2 se překrývají, a proto dvojice $\langle R_4, S_2 \rangle$ je součástí výsledku primárního filtru. MOO T odstraníme z množiny $R \cup S$, neboť již nemůže být součástí žádné dvojice, která by prošla primárním filtrem.
- 4) Pohybujeme sweep-line skrz rovinu (seřazenou množinu $R \cup S$), dokud nenarazíme na další MOO. V našem případě to bude S_2 . Ten zpracujeme postupem uvedeným v bodech 2) a 3).
- 5) Skončíme, až $R \cup S = \emptyset$.

Optimalizace MOO-spojení 2

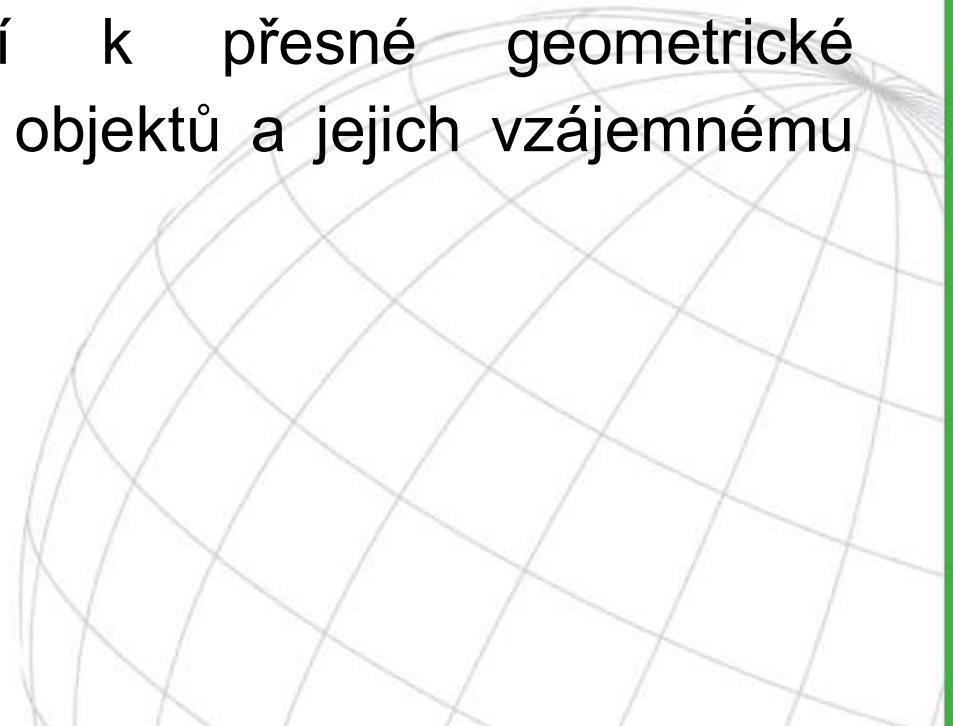
- Primárním filtrem prošly následující potenciální hity:

- $\langle R_4, S_2 \rangle$
- $\langle R_1, S_2 \rangle$
- $\langle R_1, S_3 \rangle$
- $\langle R_2, S_3 \rangle$
- $\langle R_3, S_3 \rangle$.



Prostorové spojení pomocí prostorových indexů

- Bohužel, ne všechny dvojice objektů získaných MOO-spojením jsou hity.
- Pro přesné odlišení chybných hitů se mohou použít další aproximace a kritéria a teprve v nerozhodnutelných případech se přistoupí k přesné geometrické reprezentaci prostorových objektů a jejich vzájemnému porovnání.

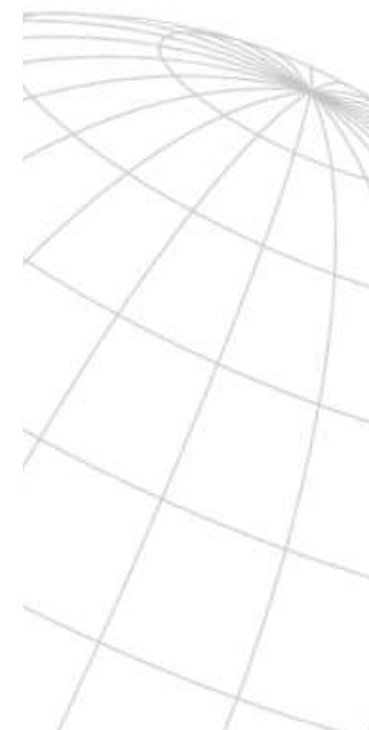
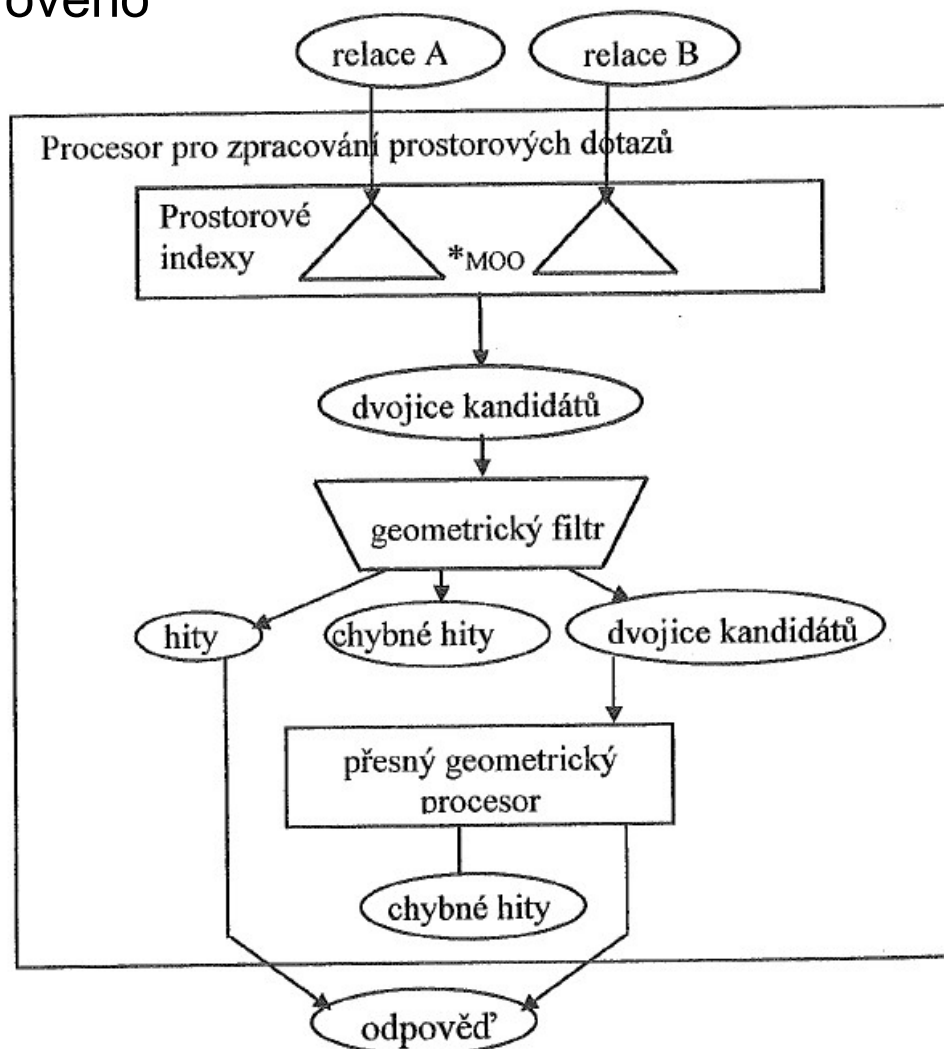


Prostorové spojení pomocí prostorových indexů

- Prostorové spojení tak lze realizovat ve třech krocích:
 - 1) Za použití indexové struktury pro organizaci aproximovaných prostorových objektů se spočítá **MOO-spojení**, které ovšem poskytne pouze množinu kandidátů na spojení objektů.
 - 2) V tzv. **geometrickém filtru** se pomocí kvalitnějších aproximací a přídavných kritérií odhalují hity, falešné hity a dvojice, které nelze rozhodnout. Tyto vstupují do třetího kroku.
 Pozn.: Geometrický filtr nemusí být v prost. databázi implementován.
 - 3) Pomocí přesné geometrie objektů (pro dvojice, které nešly rozhodnout v kroku 2) se oddělí další hity.
- Třetí krok je výpočetně nejdražší, používá algoritmů výpočetní geometrie.

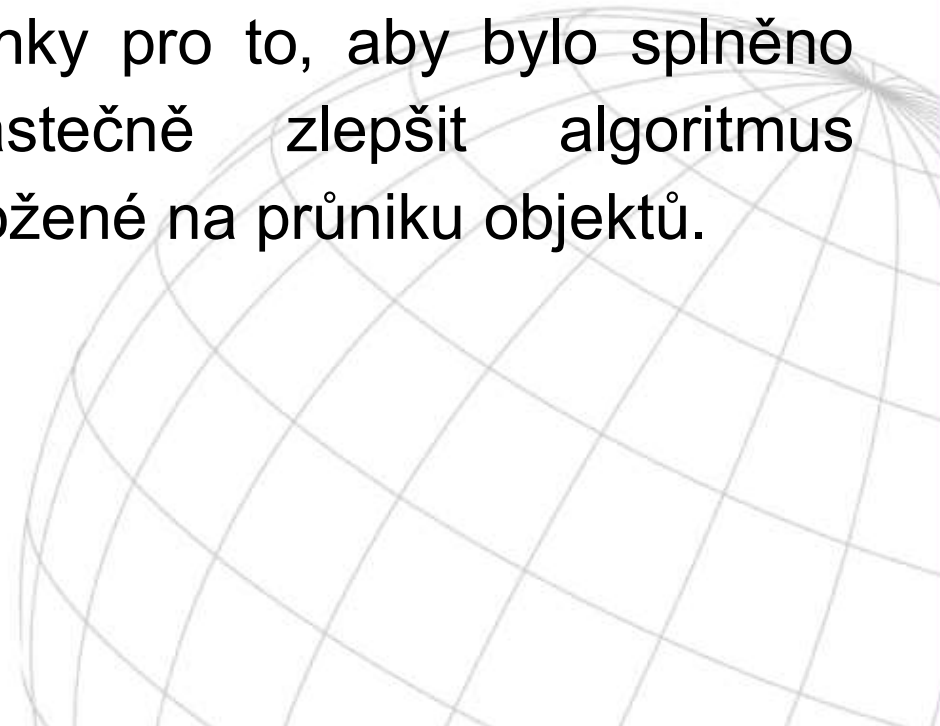
Prostorové spojení pomocí prostorových indexů

Zpracování prostorového spojení



Geometrický filtr

- Jsou-li spolu s objekty a jejich konzervativními aproximacemi ukládány i jejich **nevyužité („mrtvé“)** **plochy** nebo **progresivní aproximace**, lze jednoduše získat několik testů pro rozpoznání hitů.
- Existují postačující podmínky pro to, aby bylo splněno $\cap(o_1, o_2)$. Lze tak částečně zlepšit algoritmus pro prostorové spojení založené na průniku objektů.



Geometrický filtr

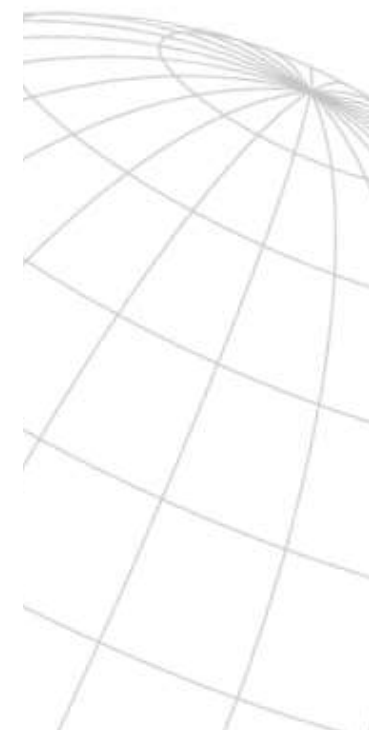
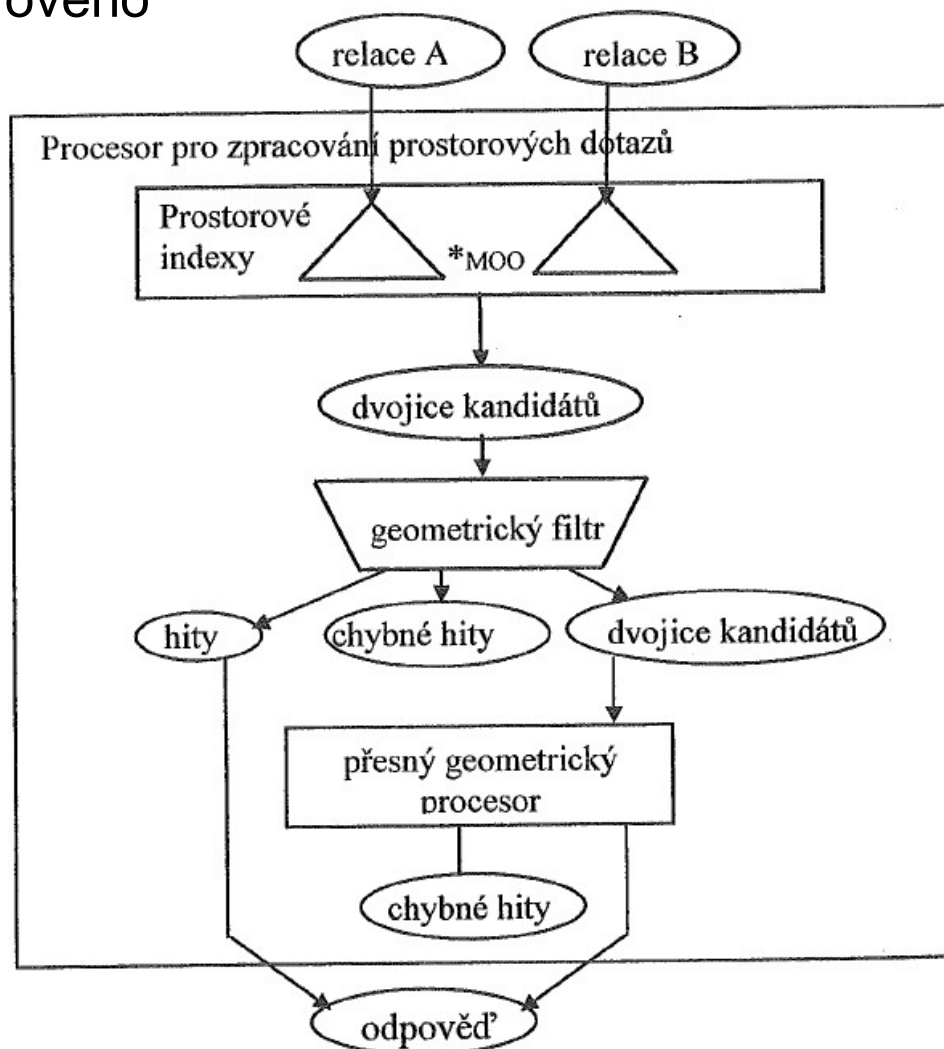
- Tvrzení (využívá jich **geometrický filtr**):

Označme K_{Apr} , resp. P_{Apr} funkci přiřazující objektu jeho konzervativní, resp. progresivní aproximaci. Označme NA funkci přiřazující konzervativní aproximaci objektu jeho nevyužitou (mrtvou) plochu.

- Jestliže $K_{Apr}(o_1) \cap K_{Apr}(o_2) > NA(o_1) + NA(o_2)$, pak $o_1 \cap o_2 \neq \emptyset$.
- Jestliže $P_{Apr}(o_1) \cap P_{Apr}(o_2)$, pak $o_1 \cap o_2 \neq \emptyset$.

Prostorové spojení pomocí prostorových indexů

Zpracování prostorového spojení



Zdroje

- JANEČKA, K.: [Zajištění konzistence prostorových dat v Informačním systému katastru nemovitostí](#). In: Proceedings of GIS Ostrava 2008. Tanger. Ostrava, 2008. s. 1-8. ISBN 978-80-254-1340-1.
- KOLINGEROVÁ, I.: [Přednášky k předmětu Vybrané algoritmické metody](#). FAV ZČU v Plzni.
- MURRAY, Ch.: [Oracle Spatial Developer's Guide, 11g Release 1 \(11.1\)](#). Oracle. 2009.
- POKORNÝ, J.: [Prostorové datové struktury a jejich použití pro indexaci prostorových objektů](#). In: Proceeding of GIS Ostrava 2000. Ostrava, 2000.
- POKORNÝ, J.: [Prostorové objekty a SQL](#). In. Proceedings of GIS Ostrava 2001. Ostrava, 2001. ISSN: 1213-239X.
- POKORNÝ, J.; ŽEMLIČKA, M.: [Základy implementace souborů a databází](#). Karolinum. Praha, 2004. 211 s. ISBN: 80-246-0837-5.
- ŽEMLIČKA, M.: [Přednášky k předmětu Organizace a zpracování dat II](#). MFF UK v Praze.

