geomatika FAV

KMA/PDB
Prostorové datové struktury a jejich použití k indexaci prostorových objektů

Karel Janečka

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

Obsah

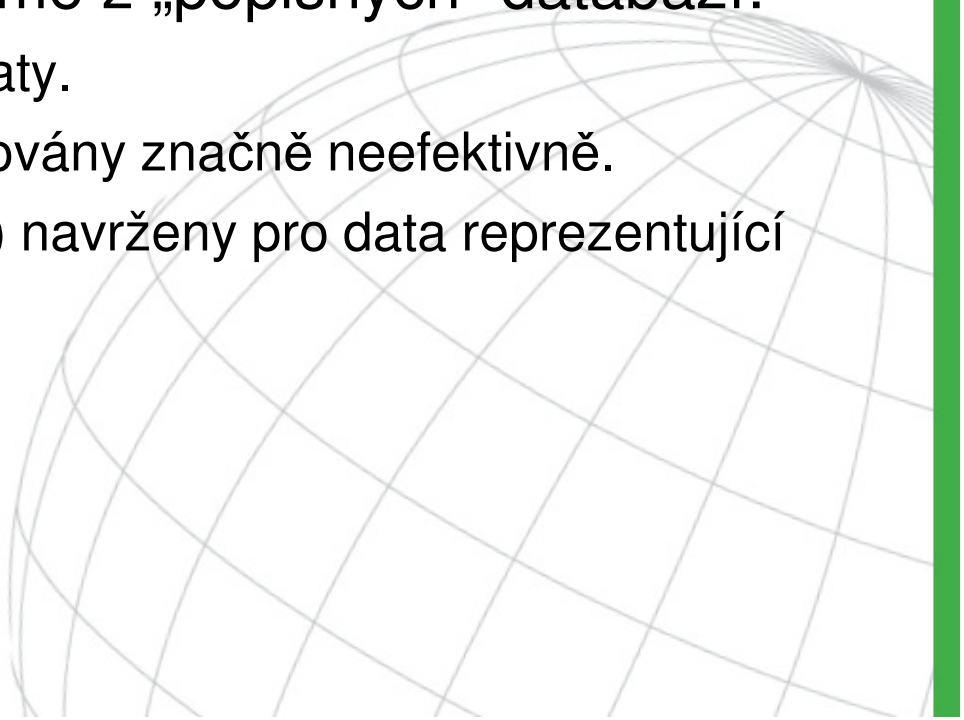
- Motivace.
- Reprezentace prostoru a prostorových objektů.
- Prostorové datové struktury (PDS) pro indexaci bodů.
- PDS pro indexaci obecných objektů.

Motivace

- **Realizace prostorových dotazů** $\Rightarrow$ použití speciálních datových struktur.
- Umožnění indexovat prostorové objekty tak, aby bylo možné uvažovat topologii prostorových objektů (zohlednění prostorových vazeb mezi objekty).
- Většina těchto struktur je založena na stromech.

Motivace

- PDS – přímo reprezentují topologické vztahy mezi prostorovými objekty, skutečné prostorové objekty tvoří až druhou vrstvu.
- Příznačné pro prostorovou indexaci je, že nelze přímo použít datové struktury známé z „popisných“ databází:
  - Problém s vícerozměrnými daty.
  - Prostorové dotazy vyhodnocovány značně neefektivně.
  - Tyto struktury (např. B-strom) navrženy pro data reprezentující body v prostoru.



Reprezentace prostoru a prostorových objektů

- PDS lze rozdělovat podle různých kritérií:
  - Jedním z možných kritérií je **pojetí výchozího prostoru *P*, ve kterém jsou prostorové objekty umístěny:**
    1) Transformace prostorových objektů do prostoru jiné dimenze, kde se jeví jako body → transformační přístup.
    2) Prostor *P* je rozdělen (nebo dynamicky rozdělován) do podprostorů.
- Přístup 1) dimenze prostoru se může:
  - zvýšit
  - snížit ⇒ linearizace prostoru
- Přístup 2) podprostory:
  - se **překrývají** (např. R-strom)
  - jsou **disjunktní** (např. k-d-strom, čtyřstrom (*Quad-tree*)).

Transformační přístup

- Prostorové objekty: obecně transformace z prostoru dimenze *k* do prostoru dimenze *2k*.
- Např. úsečka v 2D → (x1,y1,x2,y2).
- Nevýhoda:
  - Zvyšování dimenze.
  - Porušení sousedství.
- Další možností je rozdělení prostoru na buňky stejné velikosti a očíslování v souladu s nějakou vyplňující křivkou (Peanova, Hilbertova,…) – dostaneme se do prostoru nižší dimenze – **linearizace dvourozměrného prostoru**.

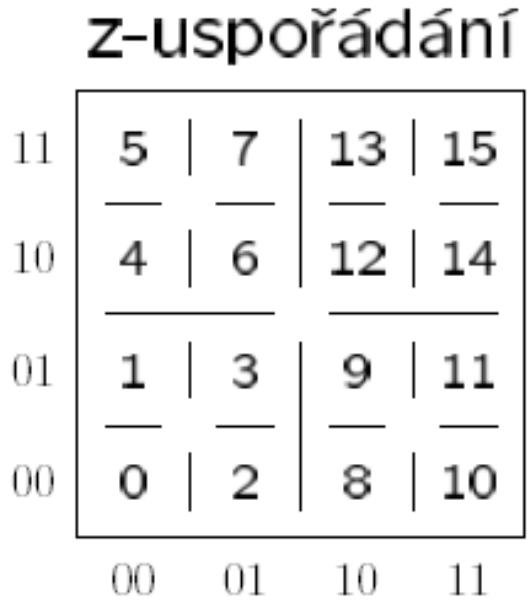
Linearizace prostoru

- z-uspořádání:
  - Reprezentace bodů [*x*, *y*], kde | *x* | = | *y* | = *w* bitů
  - Klíč je přiřazen proložením bitů:
    - ***x: x1x2…xw***
    - ***y: y1y2…yw*,** ***xi*, *yi* … *i*-tý bit *x*, resp. *y***
  - *H(x,y) = x1y1x2y2…xwyw* → **z-hodnota**
  - *H* poskytuje úplné pokrytí bodů ve čtverci.

Linearizace prostoru

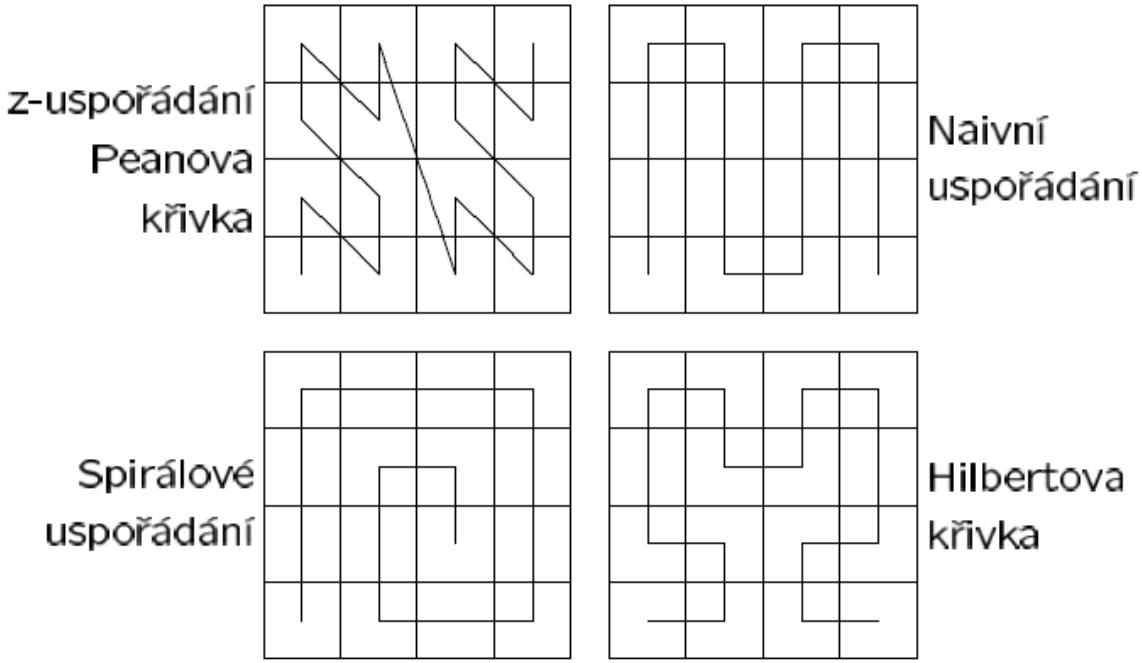
- z-uspořádání:
  - Jsou-li souřadnice buňky (x1x2, y1y2), pak (lineární) adresa buňky je x1y1x2y2.
  - Př.: Konkrétní souřadnice buňky: 1010

    Lin. adresa: 1100 (⇒12)



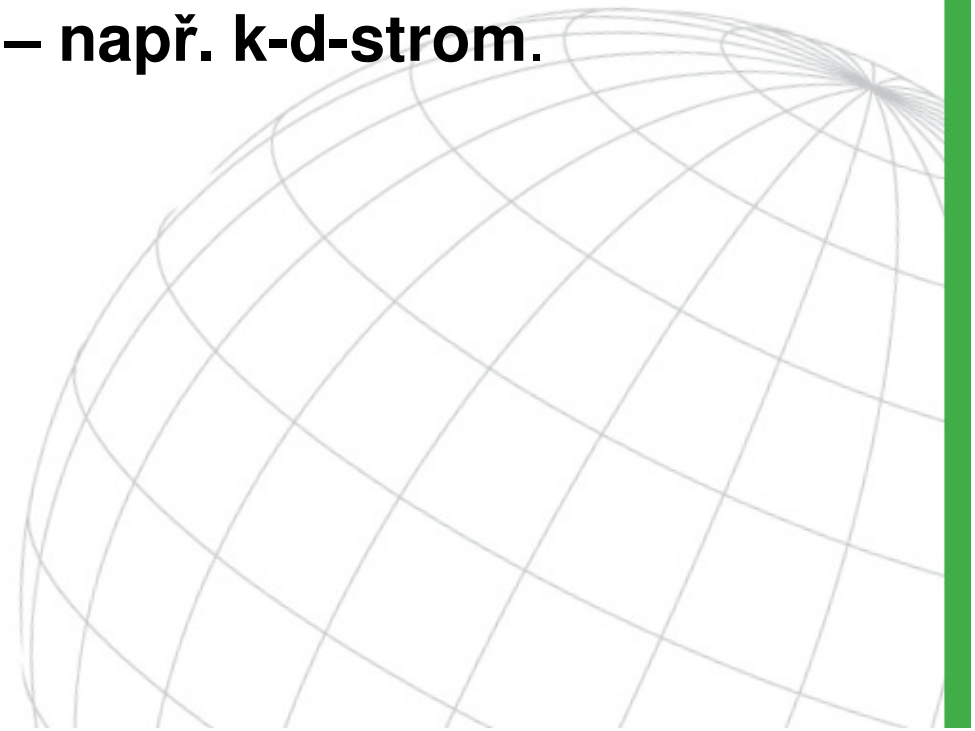
Linearizace prostoru

- Křivky vyplňující prostor:
  - **Peanova křivka – z-uspořádání**.
  - Hilbertova křivka – body, které jsou sousední v Hilbertově uspořádání, jsou sousední i v původním prostoru.
  - …



Použití nepřekrývajících se oblastí

- Prostor P je rozdělen do navzájem disjunktních podprostorů.
- Pro obecný objekt (pokud se nachází v ***s*** podprostorech) je jeho ID ***s*** *x* duplikováno – např. R+strom.
- **Vhodné pro bodová data – např. k-d-strom**.

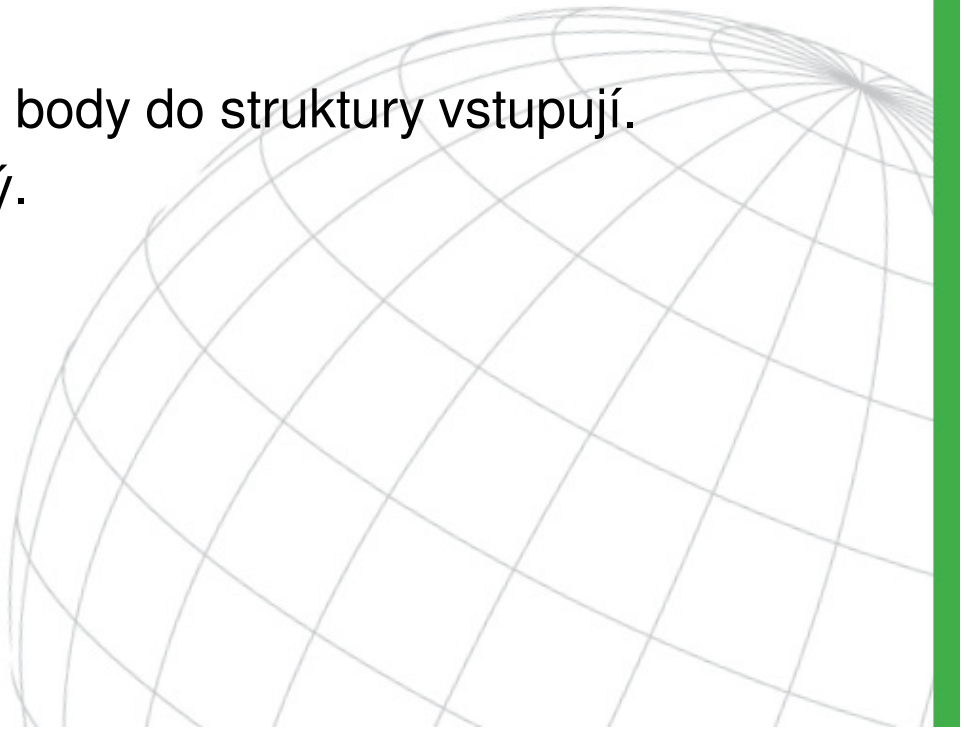


Prostorové datové struktury pro indexaci bodů

- U všech struktur, o kterých se pro indexaci bodů zmíníme, platí, že se rozděluje celý prostor **P**.
- Dlaždicový index (*Grid Tile spatial index*)
- k-d-strom.
- Čtyřstrom.
- …

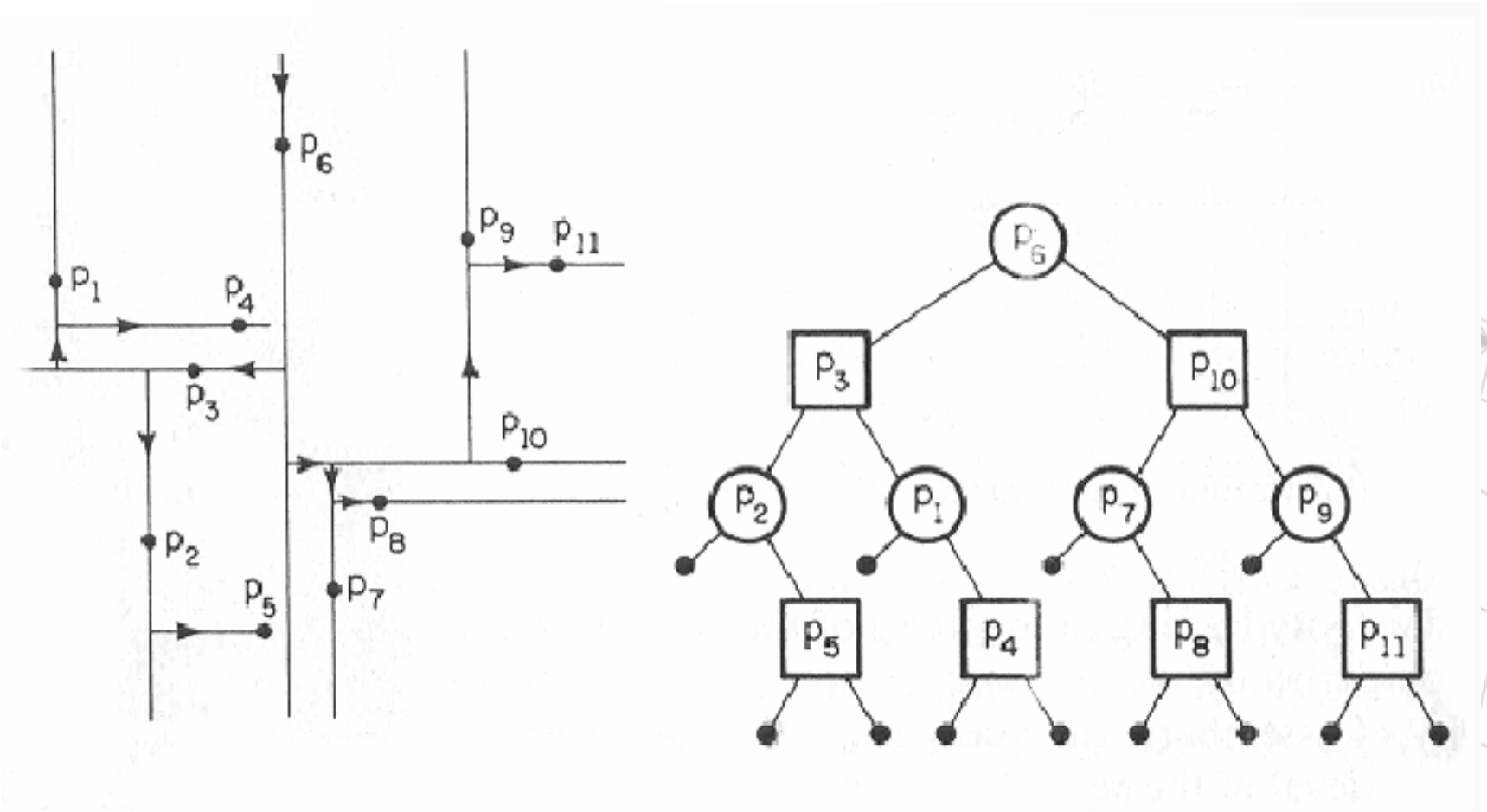
k-d-strom

- *k*-dimenzionální datová struktura, (Bentley, 1975).
- Binární strom, jehož vnitřní uzly se skládají z osy a dvou ukazatelů.
- Listy obsahují vlastní data.
- Nevýhody:
  - Citlivost na pořadí, ve kterém body do struktury vstupují.
  - Strom je výškově nevyvážený.



k-d-strom

Vstupní množina dat a odpovídající k-d-strom:

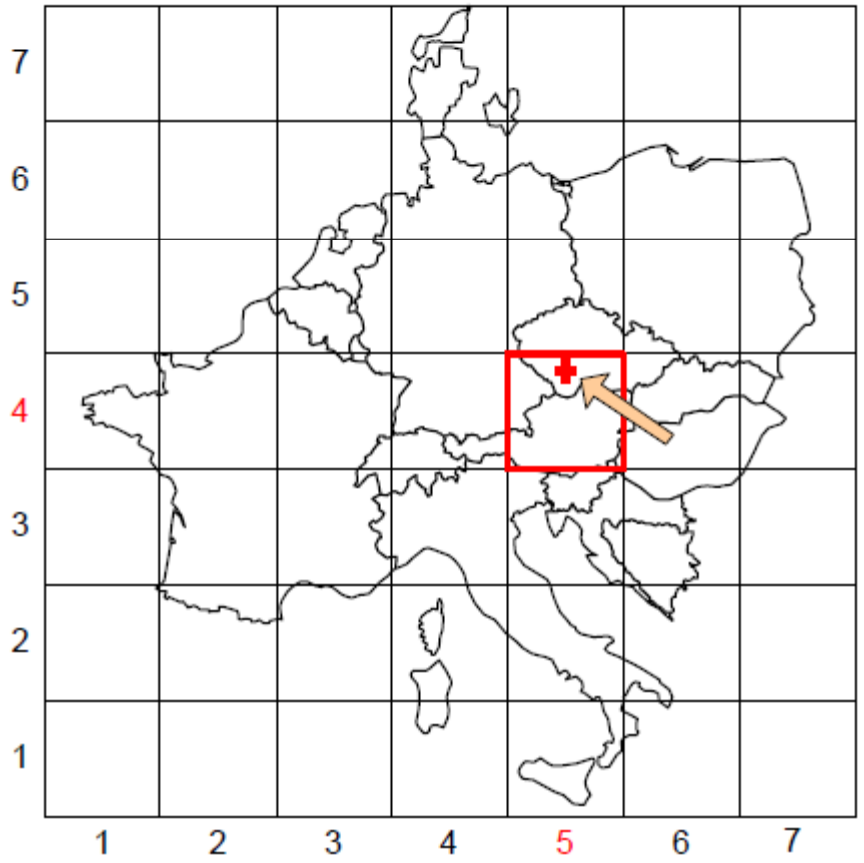


Dlaždicový index

- Tento index se skládá ze tří částí:
  - **Dlaždic**
    - představují fiktivní čtvercovou mřížku, která je vytvořena nad mapovou vrstvou v okamžiku načtení dat.
    - Nalezení správných dlaždic, které vyhovují prostorovému dotazu, je prvním krokem k jeho vyhodnocení.
  - **Obálek prvků**
    - Každý prvek má svoji obálku.
    - Tyto obálky jsou společně s dlaždicemi, do kterých zasahují, uloženy v jedné tabulce.
    - Nalezení správných obálek je druhým krokem vyhodnocení dotazu.
  - **ID prvků**
    - pro každou obálku je uloženo i ID prvku, kterému tato obálka patří.

Dlaždicový index

- Používá například ArcSDE (ESRI)
- Vyhodnocení dotazu:

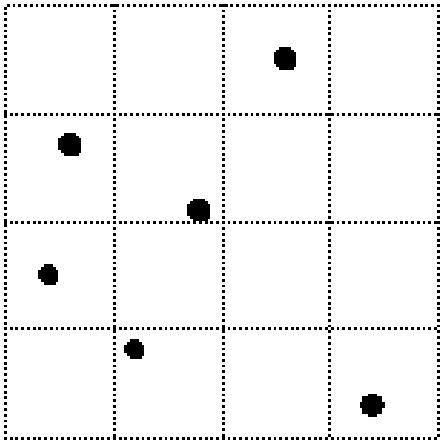


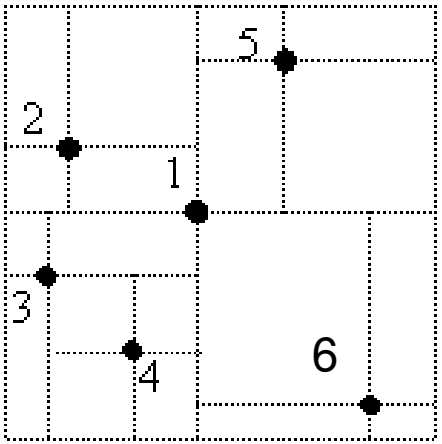
Čtyřstrom (Quad-tree)

- Finkel a Bentley (1974).
- Podporuje např. Oracle.
- Indexování bodů (lze ale i obecné objekty).
- Pro dvourozměrný prostor, který je rekurzivně dělen vždy na **čtyři podprostory**.
- Nevyvážený (nerovnoměrně rozložená data, pořadí vkládaných objektů).

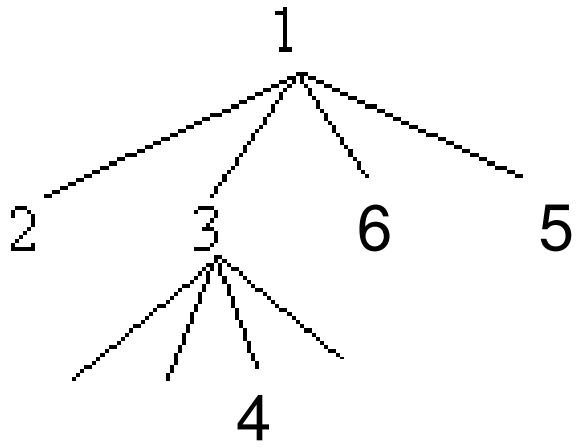
Čtyřstrom (Quad-tree)

- Vstupní množina dat (nepravidelné dělení prostoru):



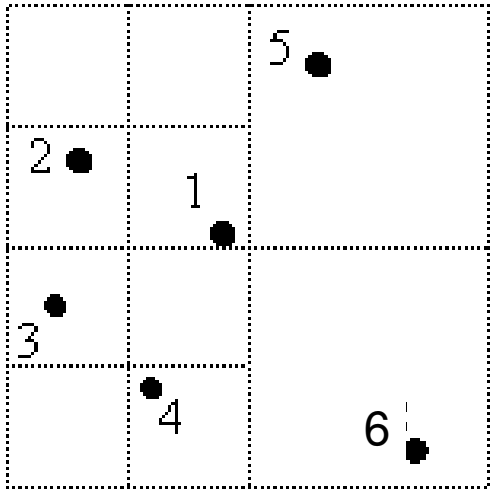


- … a odpovídající čtyřstrom (*Point Quadtree*):

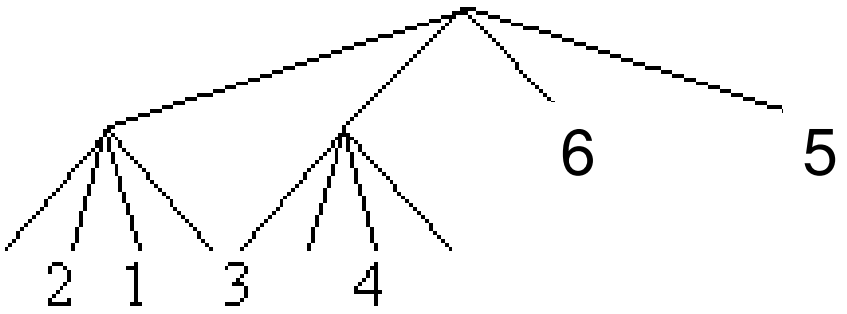


Čtyřstrom (Quad-tree)

- Vstupní množina dat (pravidelné dělení prostoru):

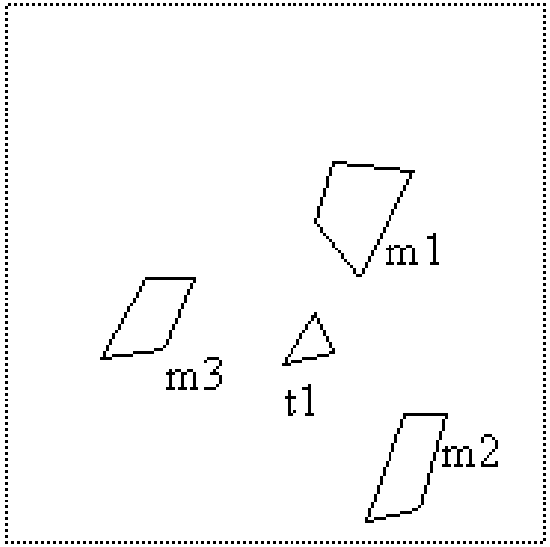


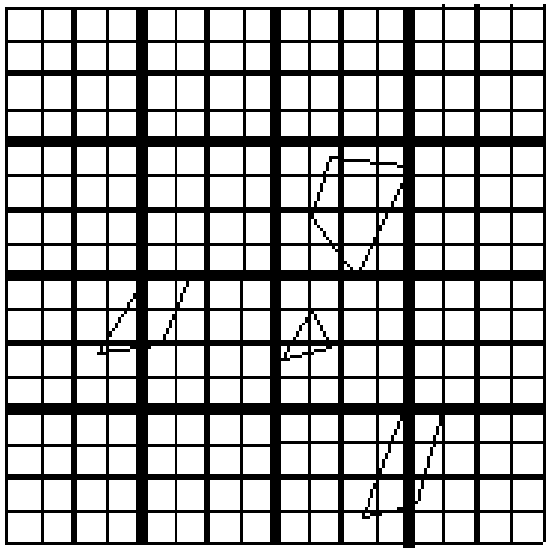
- … a odpovídající čtyřstrom (*Region Quadtree*):

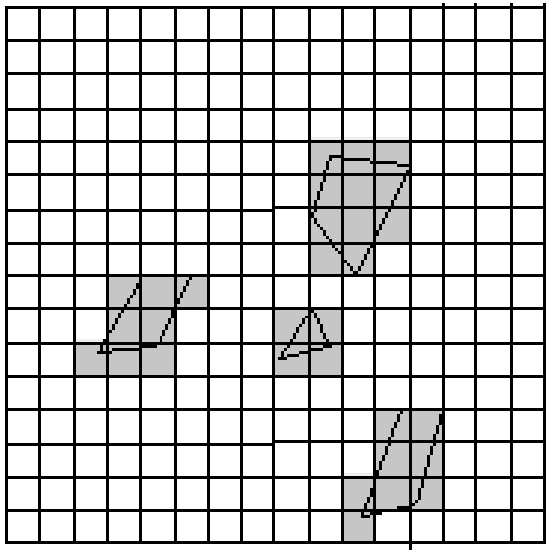


Čtyřstrom (Quad-tree)

- Použití čtyřstromu pro indexaci polygonových objektů:





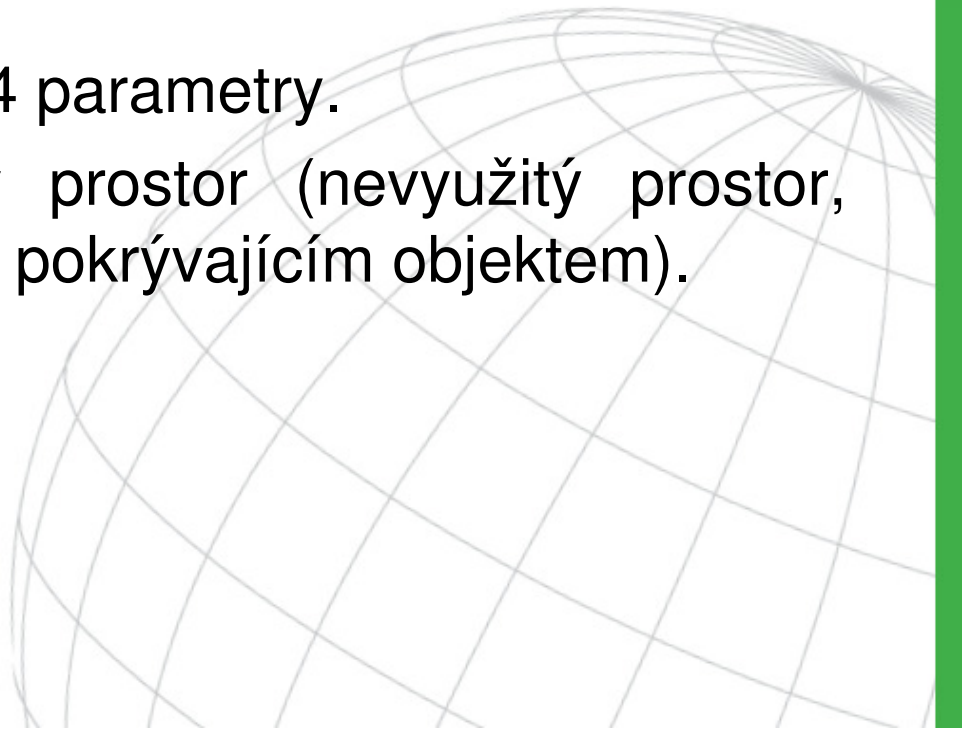


Použití pokrývajících oblastí

- Pokrývající oblasti
  - jsou konstruovány hierarchicky – většinou zdola nahoru.
  - se mohou překrývat.
  - obvykle organizovány jako stromová struktura.
- Nejznámějším představitelem tohoto přístupu je R-strom.
- **Cílem je minimalizace překrývání MOO** – získá se tím větší šance na projití méně cest ve stromě při hledání nějakého objektu.

Použití pokrývajících oblastí

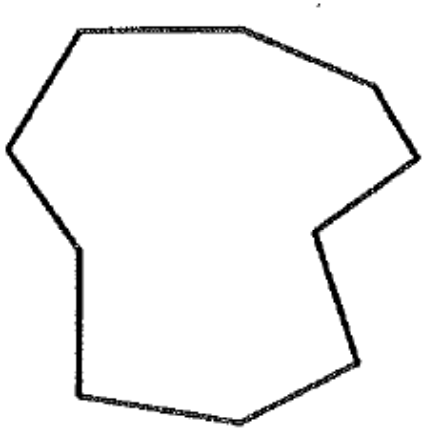
- Základní idea – každý objekt je obsažen právě v jednom z podprostorů P, který tvoří pokrývající oblast objektu.
- Pokrývající oblast – typicky **minimální ohraničující kostka** (MOK), **minimální ohraničující obdélník** (MOO) pro 2D.
- K reprezentaci MOO stačí 4 parametry.
- Problémem je tzv. mrtvý prostor (nevyužitý prostor, prostor mezi pokrývaným a pokrývajícím objektem).



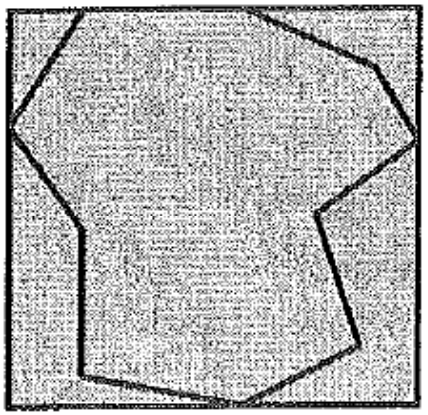
Použití pokrývajících oblastí

- Aproximace objektů mohou být i jiné než MOO.
- **Konzervativní aproximace** – každý bod objektu je bodem jeho aproximace.
  - Konvexní – větší uplatnění, existují pro ně efektivnější algoritmy.
  - Konkávní.
- MOO je příkladem konzervativní aproximace. Existují i další, většinou lepší vlastnosti než MOO, vyžadují ovšem většinou paměťově náročnější aproximaci.
- **Progresivní aproximace** – každý bod progresivní aproximace je obsažen v původním objektu.
  - Např. maximální vnořené obdélníky, maximální vnořené kružnice.

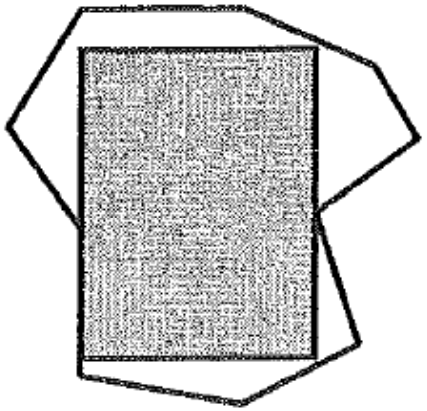
Konzervativní a progresivní aproximace



Aproximovaný objekt



Konzervativní aproximace



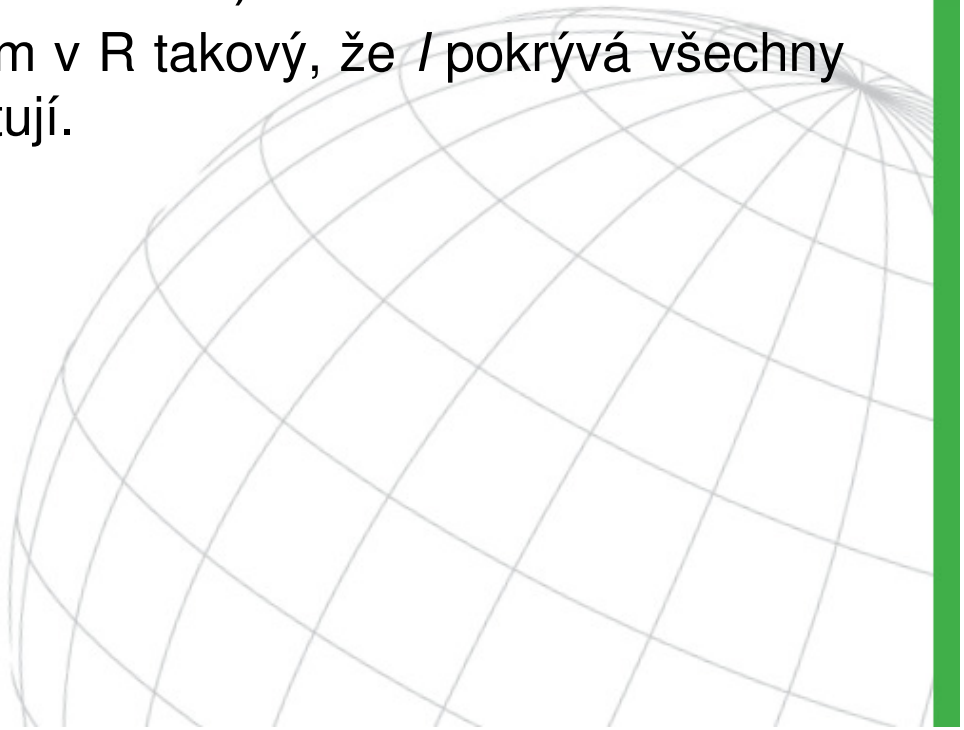
Progresivní aproximace

R-strom

- Zřejmě nejznámější prostorová datová struktura použitá k indexaci prostorových dat.
- Navržený Gutmanem v roce 1984.
- Modifikace B-stromů → záznamy v listových uzlech stromu obsahují ukazatele k datovým objektům reprezentujícím prostorové objekty.
- Technika R-stromu používá MOK/MOO.

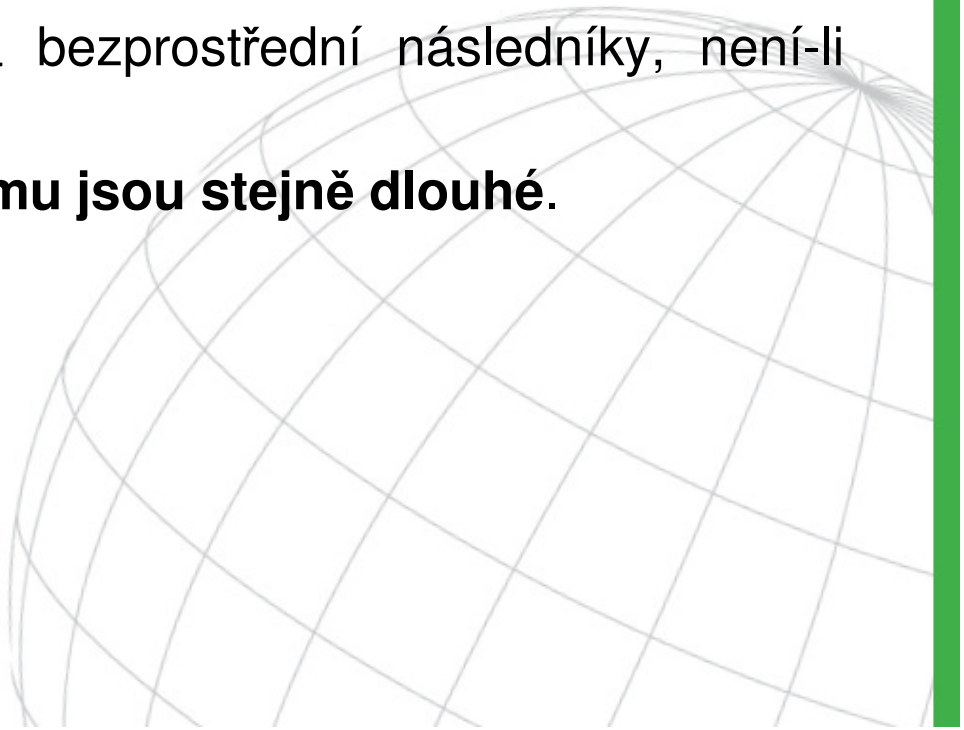
R-strom

- Struktura listu: (*I*, identifikátor objektu).
  - *I* – *d*-rozměrná kostka ohraničující prostorový objekt. Kostka má tvar (*I0*, *I1*, *I2*,…, *Id-1*), kde *Ii* je interval [*ai*, *bi*] popisující ohraničení objektu v dimenzi *i*.
- Struktura vnitřního uzlu: (*I*, ukazatel).
  - Ukazatel ukazuje na podstrom v R takový, že *I* pokrývá všechny kostky, které se v něm vyskytují.

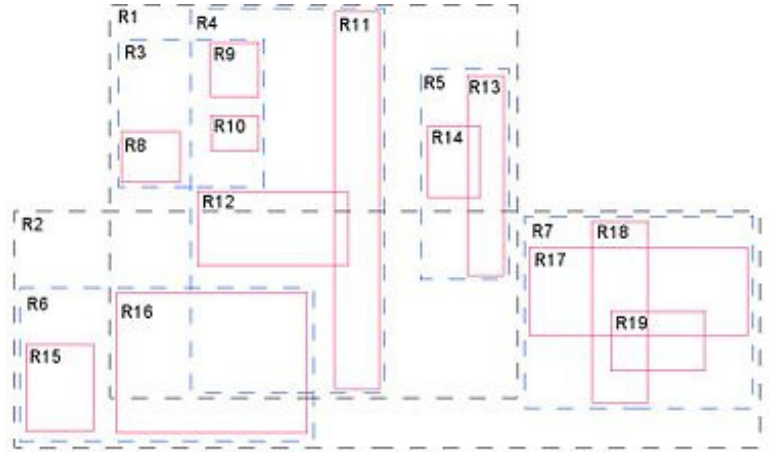


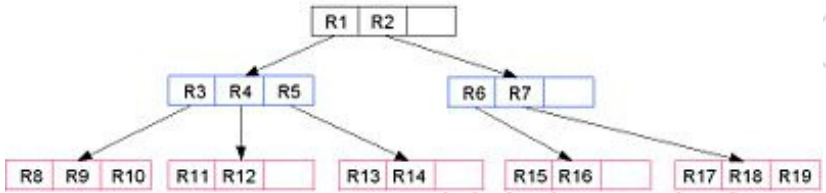
R-strom

- m-ární strom, který má následující vlastnosti:
  - Každý jeho vnitřní uzel má *n* bezprostředních následníků, $n \in \langle m1, m \rangle$. $m1 \leq m/2$
  - Každý listový uzel obsahuje *n* indexových záznamů, $n \in \langle m1, m \rangle$.
  - Kořen má nejméně dva bezprostřední následníky, není-li listem.
  - **Všechny cesty v R-stromu jsou stejně dlouhé**.



R-strom


R1
R4
R11
R3
R9
R5
R13
R10
R14
R8
R12
R2
R7
R18
R17
R6
R16
R19
R15


R1 R2
R3 R4 R5
R6 R7
R8 R9 R10
R11 R12
R13 R14
R15 R16
R17 R18 R19

R-strom

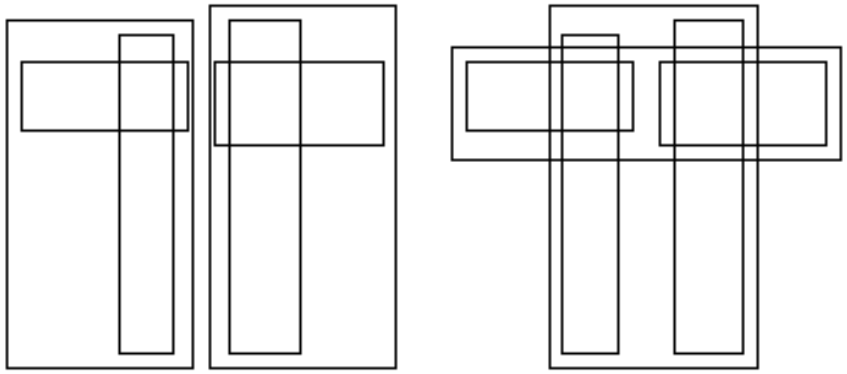
- R-strom je struktura **dynamická**, tzn. je založena na štěpení a slévání stránek.
- **Hledání v R-stromu není určeno jednou větví.** Protože vícerozměrné kostky ohraničující kostky v jednotlivých podstromech se mohou překrývat, je možné, že existuje více než jedna možnost, jak pokračovat při prohledávání stromu z jednoho uzlu.
- Tím je hledání složitější a optimalizace používané při konstrukci R-stromu jsou založeny na požadavku **co nejvíce separovat ohraničující** kostky, aby se omezil prostor vyhledávání.

R-strom

- Algoritmus vkládání indexového záznamu do R-stromu má pro zpracování dotazů zásadní důležitost.
- Je založen na strategii nalézt takovou větev ve stromě, tj. takový list R-stromu, že **opravy kostek po cestě ke kořenu povedou k co nejmenším změnám** – použití různých heuristik.
- Důležitá rovněž operace štěpení – řeší se problém, jak rozdělit neuspořádanou množinu kostek do dvou uzlů.

R-strom

- Rozdělení záznamů do dvou uzlů bude uděláno tak, aby bylo co nejméně pravděpodobné, že bude potřeba oba uzly při prohledávání zkoušet.
- Protože jsou uzly navštěvovány z uzlu-předchůdce, je nutné minimalizovat celkový objem kostek v uzlech.



špatné x dobré štěpení

R-strom

- Pro rozdělení uzlu je možné použít algoritmus uvažující všechny možnosti. Hledá se globální minimum, přičemž algoritmus má exponenciální složitost.
- Guttman – kvadratický algoritmus, který aproximuje řešení.

Kvadratický Guttmanův algoritmus
(pro štěpení uzlu R-stromu)

- Problém je rozdělen na výběr prvních kostek do uzlů a na připojování dalších. Vyberou se takové dvě první kostky (zárodky), které by měly největší mrtvý prostor, kdyby patřily do jedné skupiny.
- Výběr dalších kostek se řídí kritériem co nejmenšího přírůstku, přidáme-li kostku do skupiny. Takové kostka se vybere a do takové skupiny se přiřadí.

Kvadratický Guttmanův algoritmus
(pro štěpení uzlu R-stromu)

- Tento přístup může vést k nerovnoměrnému zaplňování obou uzlů. Z tohoto důvodu se eventuálně zbytek kostek doplní do prázdnějšího uzlu.
- V případě, že některé podmínce vyhovuje více kostek, je možné použít další heuristiky, jako:
  - Vložit záznam do uzlu s celkovým menším objemem.
  - Vložit záznam do uzlu, kde je méně záznamů.

Vylepšení štěpení uzlu R-stromu

- Jedno vylepšení R-stromů navrhl Greene:
  - Spočtení zárodků jako u Guttmanova algoritmu.
  - V každé ose se spočte vzdálenost vybraných kostek ***E****i*, ***E****j*.
  - V každé ose se normalizují vzdálenosti vydělením velikostí hrany pokrývající kostky z předchůdce uzlu.
  - Vybere se osa, kde je normalizovaná vzdálenost největší.
  - ==========
  - Setřídíme kostky ***E****i* ve vybrané ose *j* podle souřadnice ***a****j*;
  - Prvních $\lceil (m+1)/2 \rceil$ záznamů zůstane ve starém uzlu, zbytek vložíme do uzlu nového.

Ukázka štěpení přeplněného uzlu R-stromu

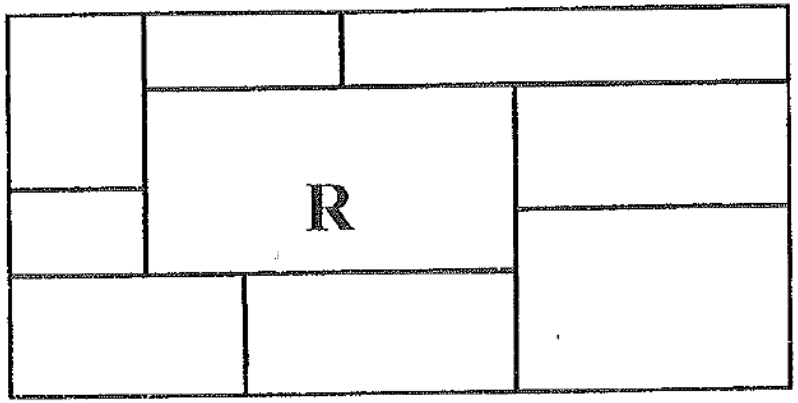
- Guttman.
- Green.

Teoretické problémy s MOO

- Mezi objektem a MOO existuje **mrtvá plocha**.
- Ta nemusí být zanedbatelná!
- Otázka – Je možné sestrojit R-strom tak, aby listové uzly obsahovaly kostky s nulovým překrytím?

Teoretické problémy s MOO

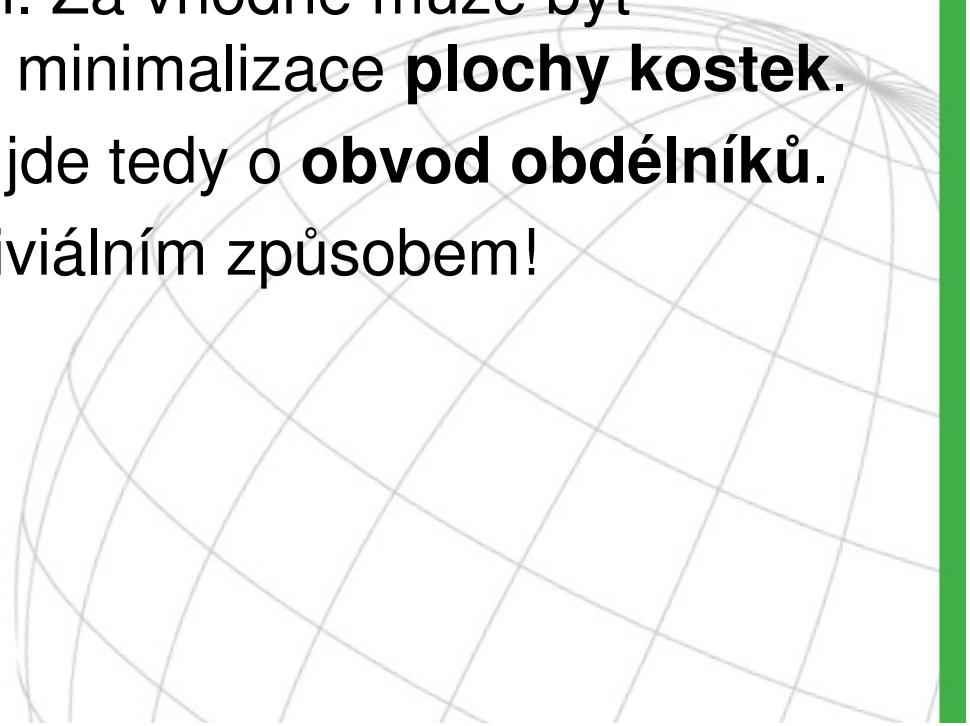
- Tvrzení (O nulovém překrytí pro oblasti): Pro libovolnou konečnou množinu disjunktních oblastí v rovině **neexistuje vždy** množina MOO taková, že platí:
  - každá oblast je přesně v jednom MOO,
  - každý MOO ohraničuje n oblastí, kde 1 < n < m,
  - průnik všech MOO je prázdný.



Nulové překrytí neexistuje

R*-strom

- Při budování R-stromu dynamickým způsobem se používaly dva parametry, které byly minimalizovány:
  - Prostor pokrývaný kostkou z vnitřního uzlu R-stromu.
  - Překrytí dvou kostek z vnitřního uzlu R-stromu.
- Existují však další možnosti. Za vhodné může být například zvoleno kritérium minimalizace **plochy kostek**.
- V dvojrozměrném prostoru jde tedy o **obvod obdélníků**.
- Parametry se ovlivňují netriviálním způsobem!



geomatika FAV

Ukázka štěpení přeplněného uzlu R*-stromu

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.