Distribuované DBS

Jaroslav Pokorný
MFF UK, Praha
pokorny@ksi.ms.mff.cuni.cz

Obsah

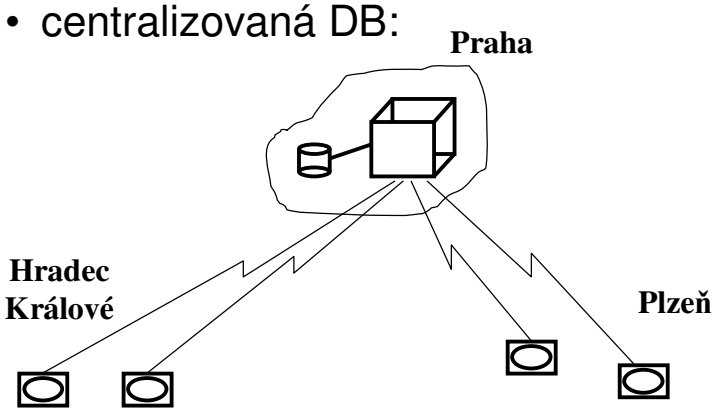
- Definice problému
- Pojem distribuovaného SŘBD
- Přístupy k DSŘBD
- Distribuované zpracování dotazu
- Transakce – zotavení z chyb

Obsah

- Definice problému
- Pojem distribuovaného SŘBD
- Přístupy k DSŘBD
- Distribuované zpracování dotazu
- Transakce – zotavení z chyb

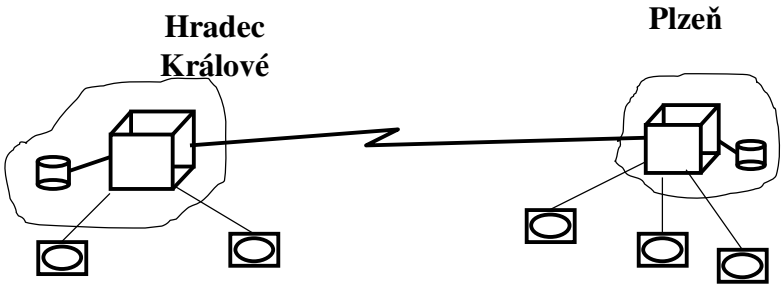
Definice problému

- centralizovaná DB:



Definice problému

- Distribuovaná DB:
- DB je umístěna na více místech
- Místa jsou propojena

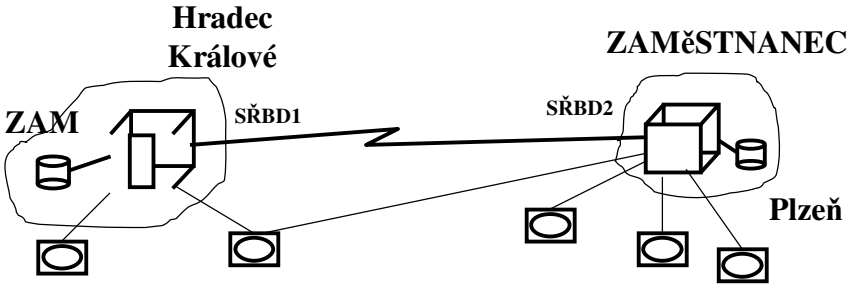


J. Pokorný - DBS2 5

Definice problému

nyní:

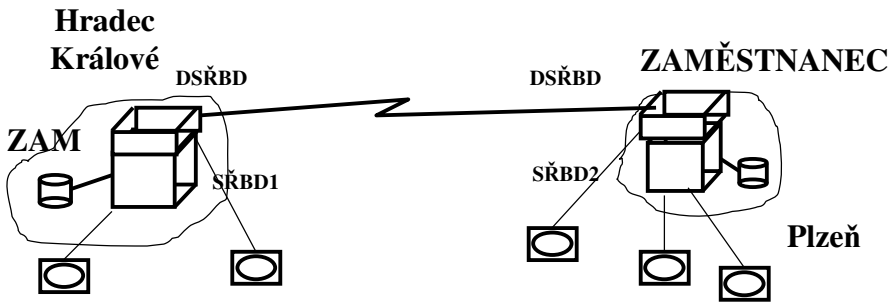
```
connect k HK; exec sql select * from ZAM; ...
connect k Pl; exec sql select * from ZAMěSTNANEC;
...
```



J. Pokorný - DBS2 6

Definice problému

lépe: **connect k distr-HK; exec sql select * from DZAM;**



J. Pokorný - DBS2 7

Obsah

- Definice problému
- Pojem distribuovaného SŘBD
- Přístupy k DSŘBD
- Distribuované zpracování dotazu
- Transakce – zotavení z chyb

J. Pokorný - DBS2 8

Distribuce

- *Distribuovaný výpočetní systém* (DVS)
  - rozložen do uzlů (míst) sítě svázaných komunikačními kanály,
  - v uzlech je umožněno autonomní uložení a zpracování dat,
  - prostředky v uzlech mohou být nehomogenní, avšak rovnocenné
  - uživatel nic neví o existenci ostatních uzlů sítě (transparence)
- *DSŘBD* je speciálním případem DVS

J. Pokorný - DBS2 9

Výhody DDBS

- data tam, kde se nejvíce používají
- rozdělení zátěže
- snižuje se cena komunikace
- větší spolehlivost systému
- lepší škálovatelnost
- lepší dostupnost, možnost sdílení, integrace
- zachování autonomního zpracování

J. Pokorný - DBS2 10

Nevýhody DDBS

- složitější návrh
- složitější optimalizace
- složitější slovník dat
- složitější globální transakční zpracování
- složitější uváznutí
- redundance implikuje složitější zotavení z chyb
- heterogennost v uzlech implikuje složitější integraci

Obsah

- Definice problému
- Pojem distribuovaného SŘBD
- Přístupy k DSŘBD
- Distribuované zpracování dotazu
- Transakce – zotavení z chyb

Typy DDBS

- shora dolů (společně organizovaná data, globální schéma)
- zdola nahoru (integrace heterogenních dat, globální schéma)
- federativní integrace (schémata importu a exportu

Pz.: speciální případy – klient/server, replikační přístup

J. Pokorný - DBS2 13

Přístupy k DSŘBD

Ve třech dimenzích:

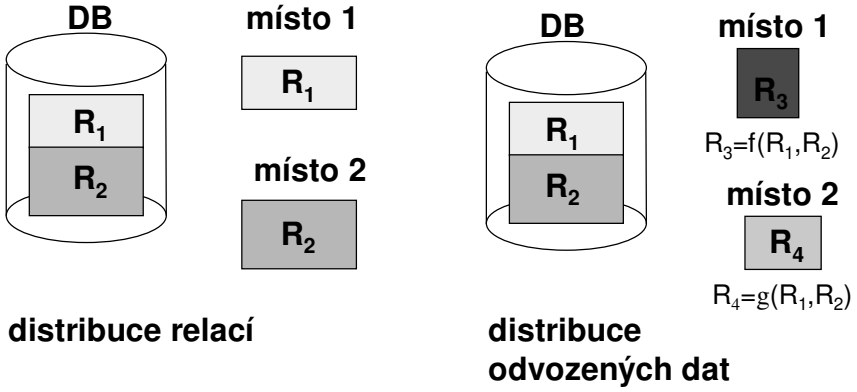
- autonomie
- transparence distribuce
- heterogennost

J. Pokorný - DBS2 14

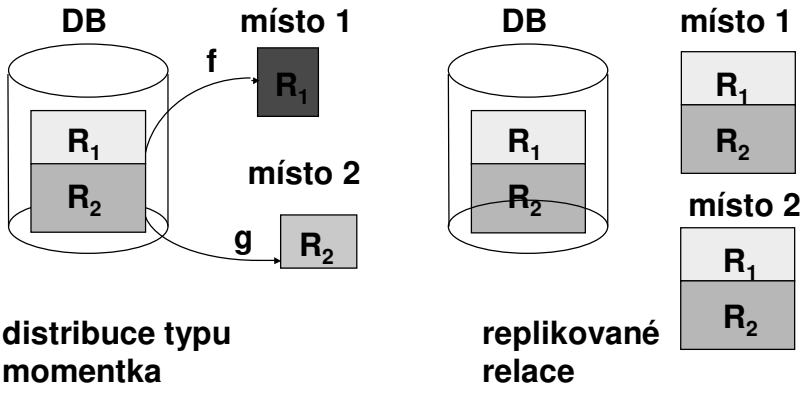
Přístupy k DSŘBD: autonomie

- těsné spojení
- federace
- úplná izolace

Přístupy k DSŘBD: transparence distribuce



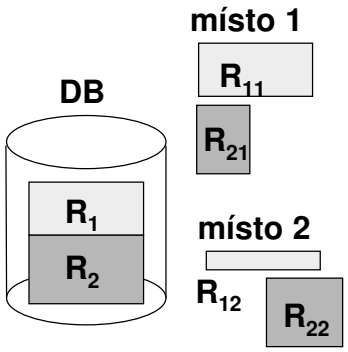
Přístupy k DSŘBD: transparence distribuce



distribuce typu momentka

replikované relace

Přístupy k DSŘBD: transparence distribuce



$R_1 = R_{11} \cup R_{12}$

$R_2 = R_{21} * R_{22}$

fragmentace relací

Pozor! Bezeztrátovost spojení.

Fragmentace

vertikální fragmentace

ROD_Č	JMÉNO	ADRESA
480423/016	Novák	Široká 24
...	...	...
430122/116	Dvořák	Kožená 2

Horizontální fragmentace

J. Pokorný - DBS2 19

Fragmentace

Definice fragmentů pomocí RA

Př.: KINO(NÁZEV_K, ADR, JMÉNO_V, Č_TELEFONU)

Fragmentace 1:

$KINO_1$:= KINO(ADR = ‘Praha 1’)

$KINO_2$:= KINO(ADR = ‘Praha 2’)

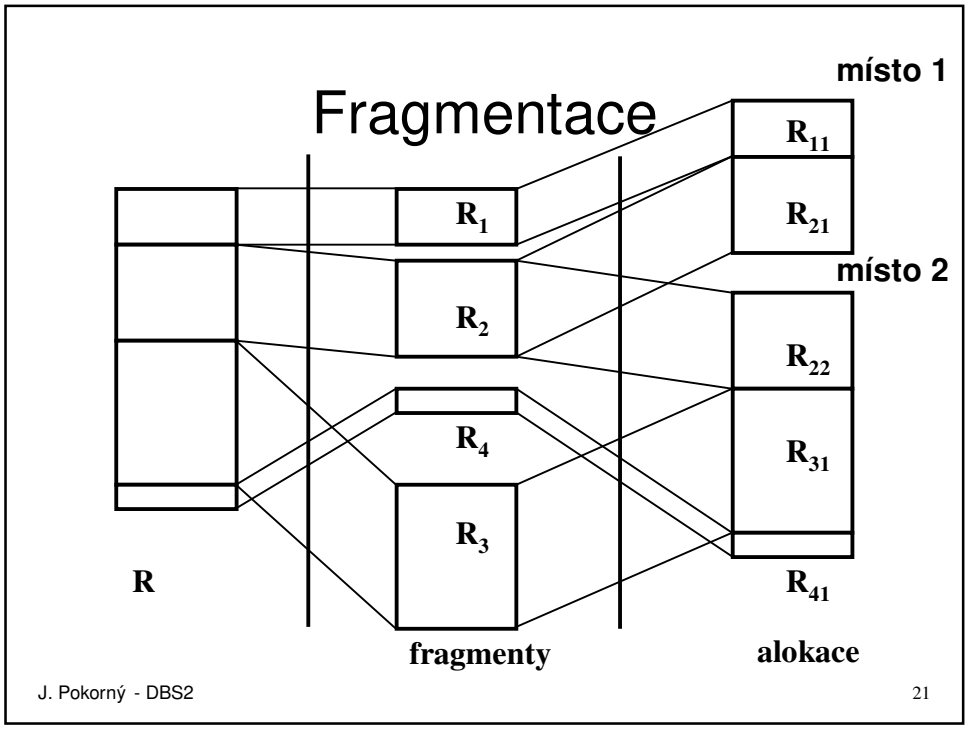
$KINO_3$:= KINO(ADR ≠ ‘Praha 1’ AND ADR ≠ ‘Praha 2’)

Fragmentace 2:

$PŘEDSTAVENÍ_1$:=PŘEDSTAVENÍ <* KINO(ADR = ‘Praha 1’)

$PŘEDSTAVENÍ_2$:=PŘEDSTAVENÍ <* KINO(ADR = ‘Praha 2’)

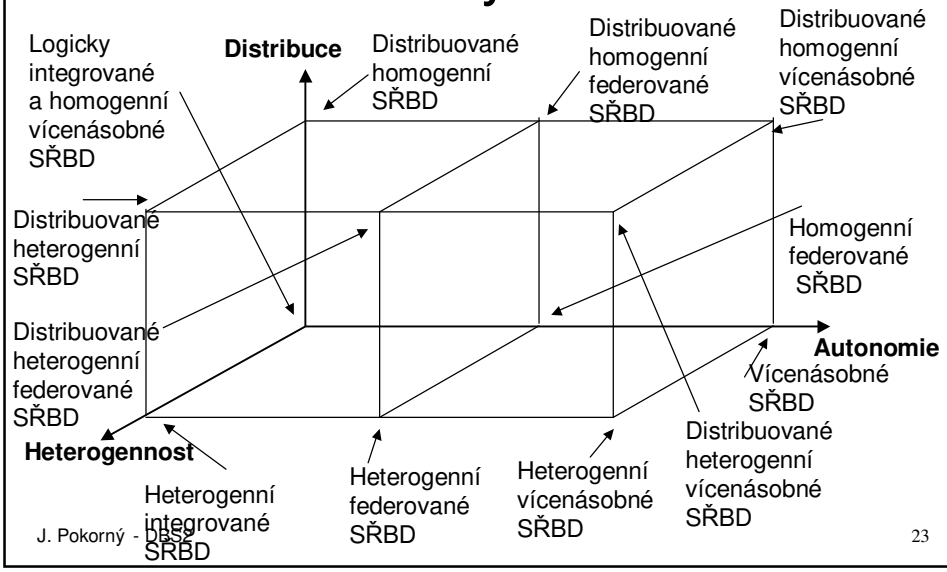
J. Pokorný - DBS2 20



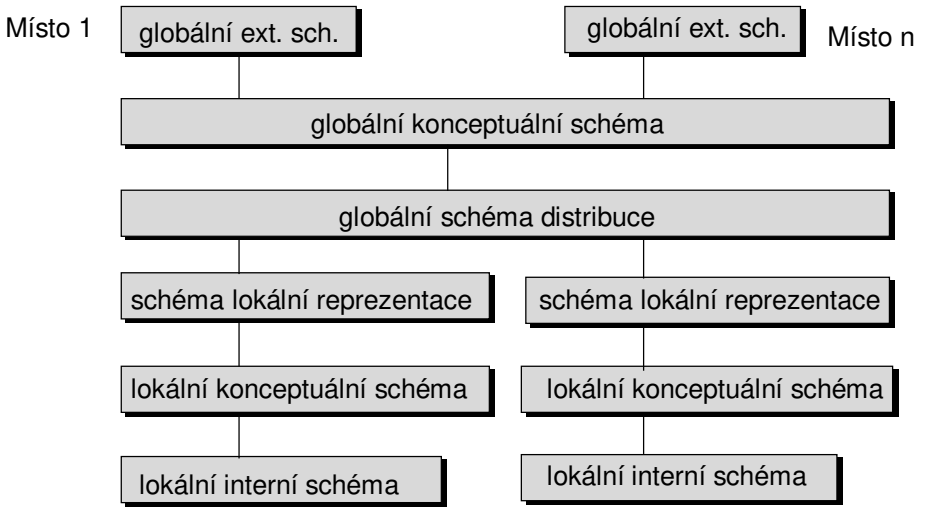
Alternativy DSŘBD

- Distribuce: shora dolů, integrace existujících DB
- Autonomie: těsné spojení, federace, úplná izolace
- Heterogennost: ANO, NE

Alternativy DSŘBD



Architektura schémat



Příklad

1. UZEL:

DÍL(Č_DÍLU, Č_DODAVATELE,CENA)

ADRESÁŘ (Č_DODAVATELE, MĚSTO)

DODAVATEL(Č_DODAVATELE, JM_DODAVATELE)

2. UZEL:

KUS(Č_KUSU, CENA, FIRMA_DODAVATELE)

DODAVATEL(Č_DODAVATELE, FIRMA_DODAVATELE, PSČ, MĚSTO)

IO: v 1. uzlu díly od dodavatelů 000-700, v 2. uzlu od dodavatelů 650-999. DÍL je ve 3NF.

Příklad

- Schémata relačních DB představují LKS v uzlech
- Strategie integrace (SLR):
  - do globálního světa „co nejvíce“ (použité NULL)
  - do globálního světa „spíše průnik dílčích světů“

1. uzel

LR_DÍL(ČÍS_DÍLU, ČÍS_DODAVATELE, CENA),

LR_ADRESÁŘ(ČÍS_DODAVATELE, MĚSTO)

2. uzel

LR_DÍL≡(KUS * DODAVATEL)[Č_KUSU, Č_DODAVATELE, CENA]

LR_ADRESÁŘ ≡ DODAVATEL[Č_DODAVATELE, MĚSTO]

Příklad

- GSD se bude obsahovat $SLR_1 \cup SLR_2$.
- Rozdělení dat v GSD bude dáno tabulkou:

LR_DÍL	1	ČÍS_DÍLU < 650
LR_DÍL	1,2	650 ≤ ČÍS_DÍLU ≤ 700
LR_DÍL	2	ČÍS_DÍLU > 700

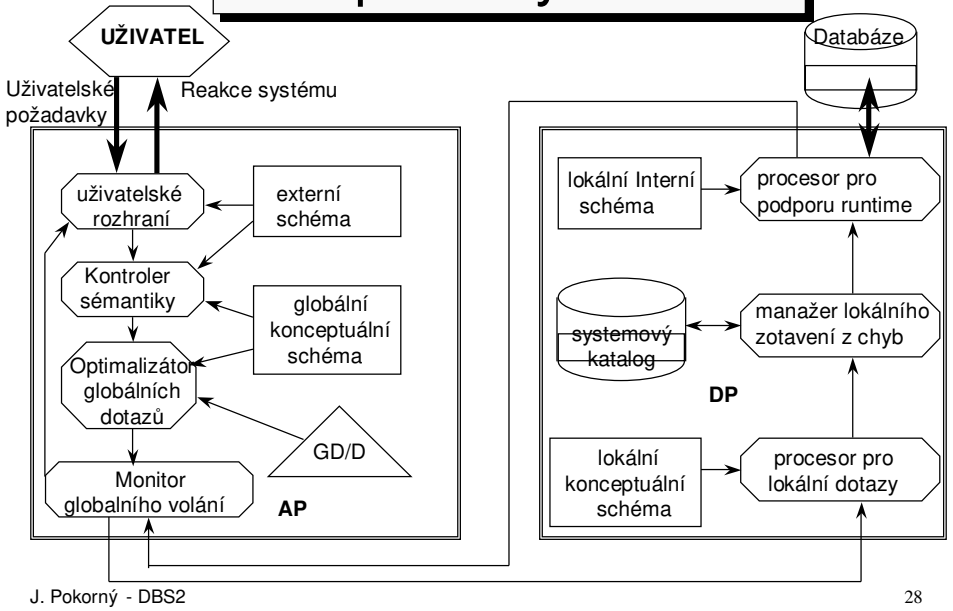
- GKS obsahuje relace GK_DÍL a GK_ADRESÁŘ se stejnými atributy jako LR_DÍL a LR_ADRESÁŘ. Může však být def. pomocí RA z GSD.

Pz.: RA je vhodná, ale nestačí, např. pro ADRESA a atributů (PSČ, MĚSTO, ULICE, Č_DOMU) je třeba speciální funkci (konkatenace)

nebo: barva (číslo vs.název)

J. Pokorný - DBS2 27

Komponenty DSŘBD



J. Pokorný - DBS2 28

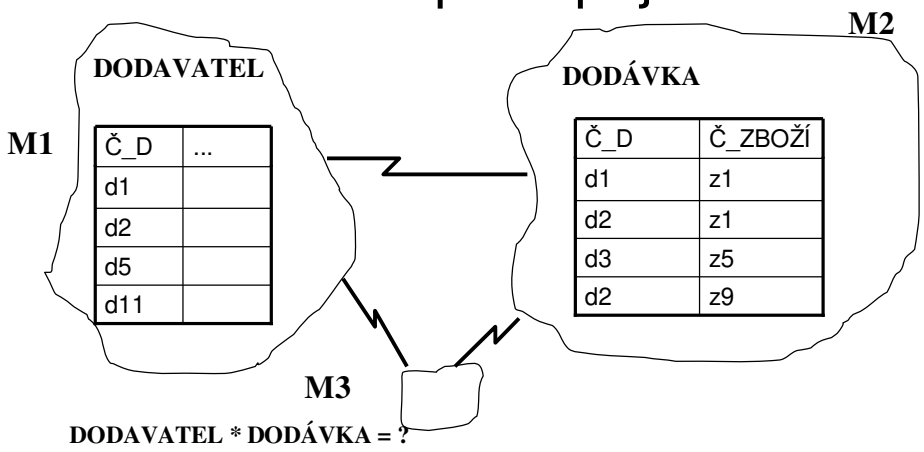
Obsah

- Definice problému
- Pojem distribuovaného SŘBD
- Přístupy k DSŘBD
- Distribuované zpracování dotazu
- Transakce – zotavení z chyb

Distribuované zpracování dotazu

- problémy (přídavné k centralizovanému případu)
  - cena přenosů
    - cpu, disk, #přenášených Byte, #přenášených zpráv
  - paralelismus / překrývání prodlev
    - Jak minimalizovat uběhnutý čas?
    - Nebo jak minimalizovat spotřebu zdrojů?

Optimalizace distribuovaného dotazu – polospojení

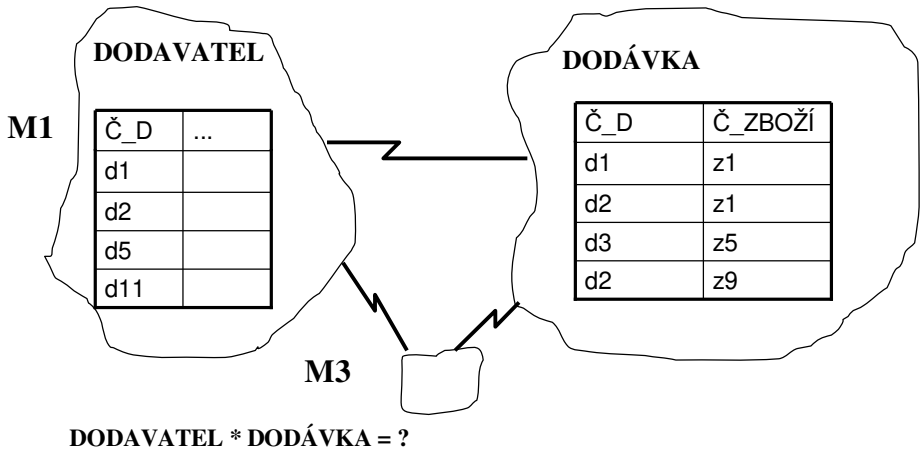


Polospojení

volba plánů:

- P1: přesuň DODÁVKA -> M1; *; přesuň výsledek -> M3
- P2: přesuň DODÁVKA->M3; přesuň DODAVATEL->M3; *
- ...
- další?

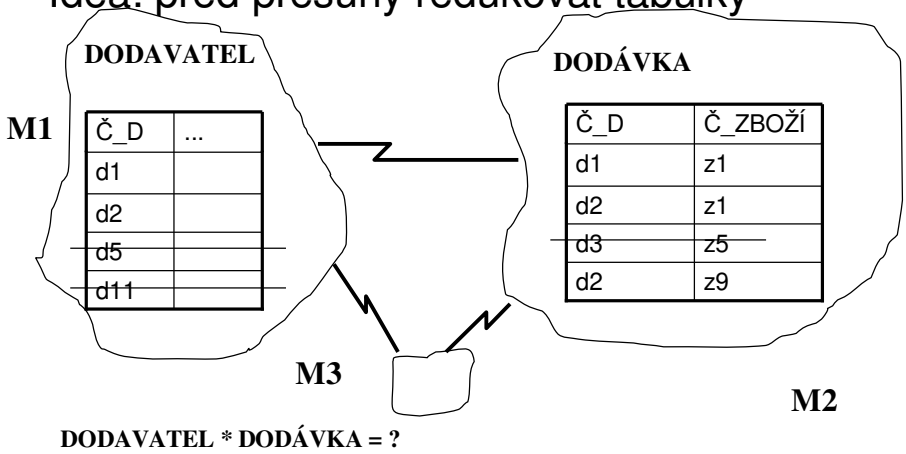
Optimalizace distribuovaného dotazu – polospojení



DODAVATEL * DODÁVKA = ?

Polospojení

Idea: před přesuny redukovat tabulky



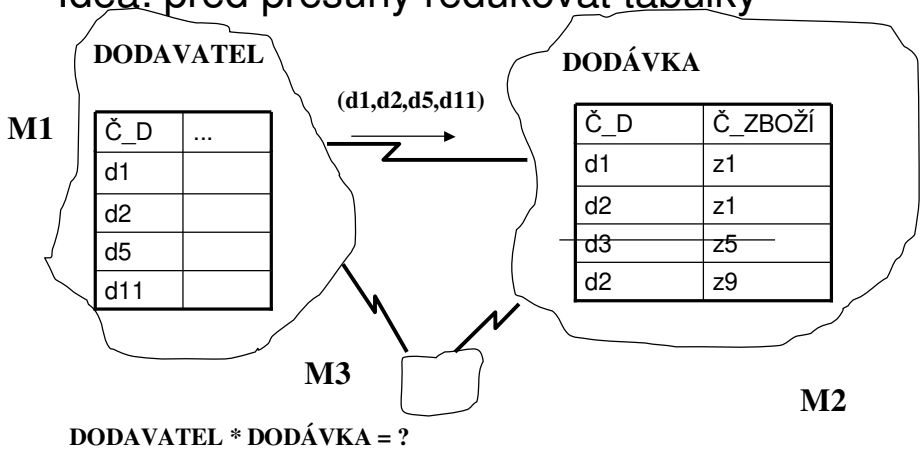
DODAVATEL * DODÁVKA = ?

Polospojení

- Jak redukovat levně?
  Př.: redukce DODÁVKA

Polospojení

Idea: před přesuny redukovat tabulky



DODAVATEL * DODÁVKA = ?

Polospojení

Formálně:
DODÁVKA' = DODÁVKA <* DODAVATEL

Pz: vyjádření polospojení pomocí RA:
R <* S ≅ (R * S)[R]

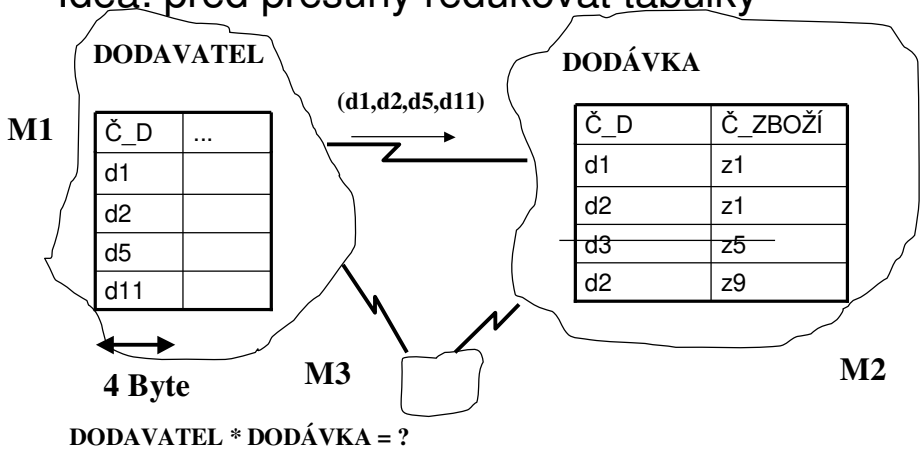
Polospojení – příklad

Velikost hodnoty každého atributu je 4 Byte

Q: cena přenosu (#Byte) pro polospojení
DODÁVKA' = DODÁVKA <* DODAVATEL

Polospojení

Idea: před přesuny redukovat tabulky



Polospojení – příklad

Předpoklad: hodnota každého atributu 4 Byte

Q: cena přenosu (#Byte) pro polospojení

DODÁVKA' = DODÁVKA <* DODAVATEL

Zde: Q = 4*4 Byte

Strategie

krok1: navrhni plán s polospojením(i)

krok2: odhadni jeho cenu (# přenesených Byte)

Polospojení – příklad

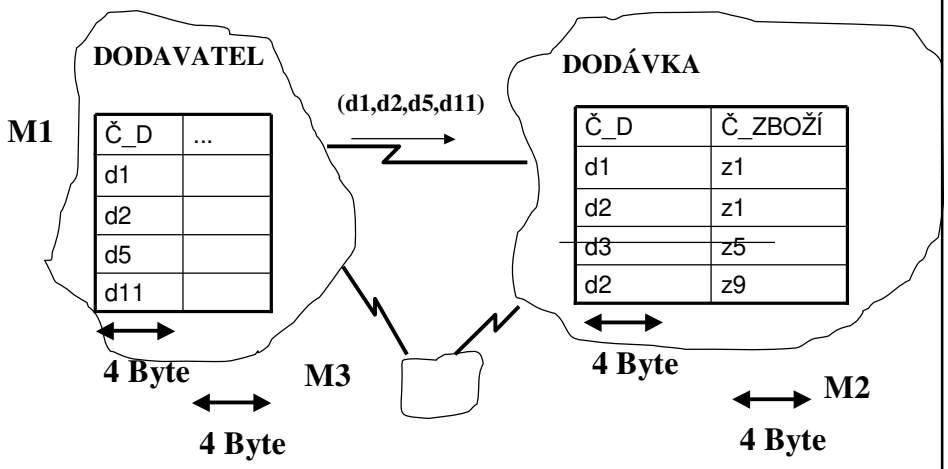
P3:

- redukuj DODÁVKA na DODÁVKA'
- DODÁVKA' -> M3
- DODAVATEL -> M3
- proveď * v M3

Cena?

Předpoklad: hodnota každého atributu 4 Byte

Polospojení



Polospojení – příklad

- Q pro P3:
  - 4*4 Byte - redukuj DODÁVKA na DODÁVKA'
  - 3*8 Byte - DODÁVKA' -> M3
  - 4*8 Byte - DODAVATEL -> M3

72 Byte celkem

Další plány?

P4:

- redukuj DODÁVKA na DODÁVKA'
- redukuj DODAVATEL na DODAVATEL'
- DODÁVKA' -> M3
- DODAVATEL' -> M3

J. Pokorný - DBS2 45

Další plány?

P4:

- redukuj DODAVATEL na DODAVATEL'
- DODAVATEL' -> M2
- proveď * v M2
- přesuň výsledek -> M3

J. Pokorný - DBS2 46

Obsah

- Definice problému
- Pojem distribuovaného SŘBD
- Přístupy k DSŘBD
- Distribuované zpracování dotazu
- Transakce – zotavení z chyb

Transakce v DSŘBD

- Problém: transakce provádí převod
  1000 Kč z HK -> 500 do Pl, 500 do ČB
- 3 dílčí transakce na 3 systémech
- Jak garantovat atomicitu (všechno nebo nic)?
- Pozorování: další typy chyb (připojení servery, prodlevy, time-outs ....)

Modul řízení transakcí

Modul řízení transakcí (kordinátor transakcí, transakční minitor)

- analyzuje požadavky T,
- rozděluje ji na dílčí transakce (DT),
- provádí syntézu výsledků,
- v DSŘBD jich může být více.

Modul řízení transakcí

Role:

- Zajišťuje přidělení privátního prostoru pro T, kterou řídí:
- Zajišťuje promítnutí hodnot do DB
- Zajišťuje řízení kopií objektu v DB
- Zajišťuje konzistenci DB

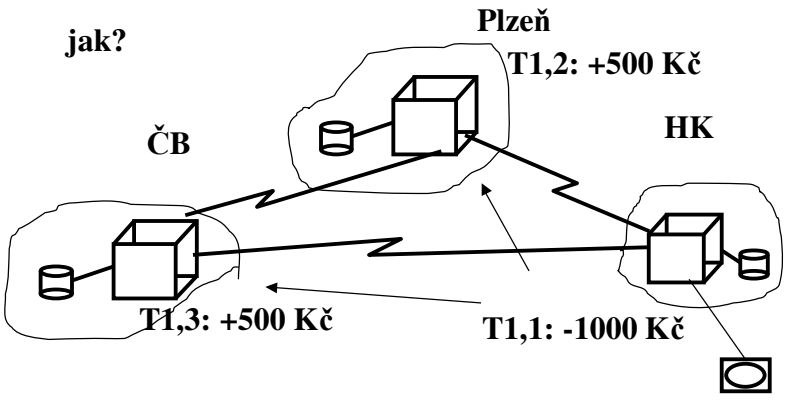
Modul řízení transakcí

akce při

- read(X): *if* B obsahuje X *then* X→T else vyber kopii X a pošli požadavek do lokálního SŘBD S, který pak umístí X do B.
- write(X): *if* B obsahuje X *then* proveď aktualizaci všech jeho kopií *else* vytvoř kopii X s danou hodnotou.
- Požadavek na EOT. Provede akce např. podle 2PC.

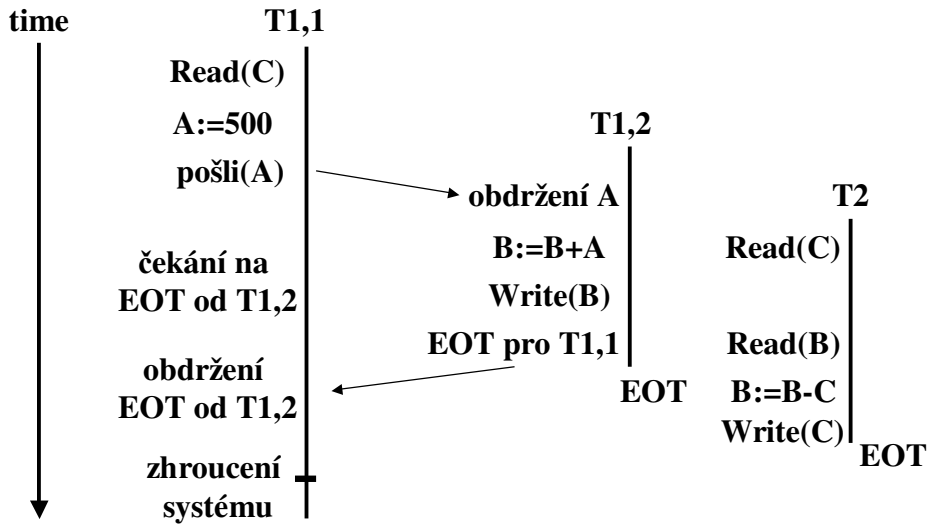
J. Pokorný - DBS2 51

Distribuované zotavení z chyb

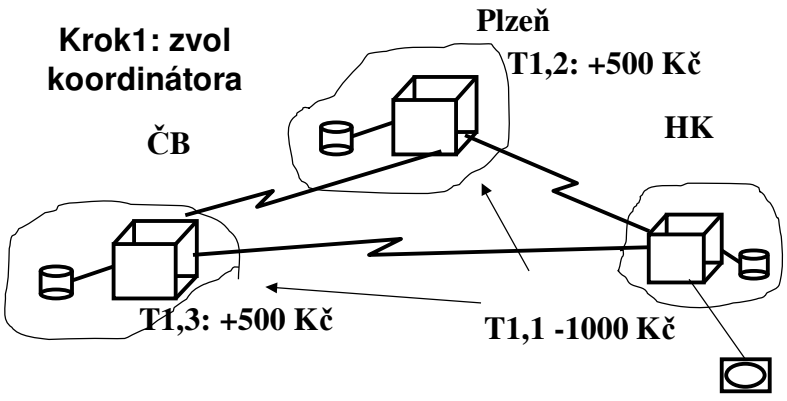


J. Pokorný - DBS2 52

Problém: dominový efekt



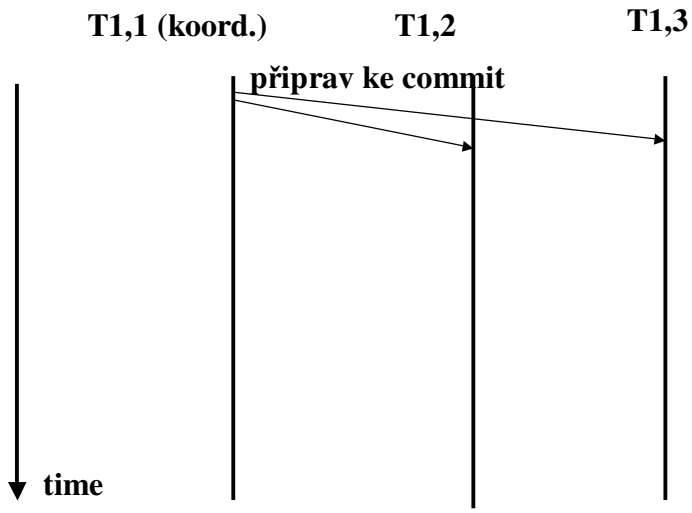
Distribuované zotavení z chyb



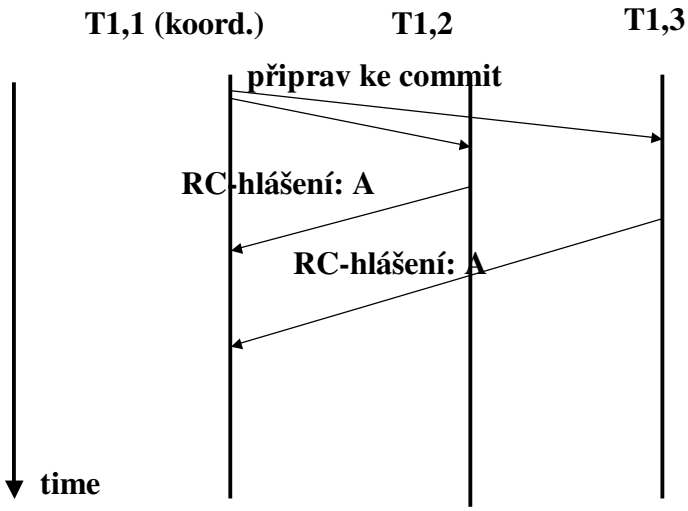
Distribuované zotavení z chyb

- Krok 2: vyvolej protokol, např., dvoufázový potvrzovací protokol (2PC)
  - Fáze 1: dosáhne-li DT RC-bod (ready-to-commit), hlásí to koordinátorovi, ale neuvolní změny pro jiné transakce. V RC-hlášení: A (je připraven k potvrzení, N (ABORT);
  - Fáze 2: koordinátor shromáždí všechna RC-hlášení. Vyšle COMMIT, když jsou všechna A. Vyšle ABORT, jestli alespoň 1 DT hlásí N nebo vypršel time-out.
  - Pz: Někdy ještě čeká na COMMIT, ABORT od všech DT.

2PC: úspěšné provedení

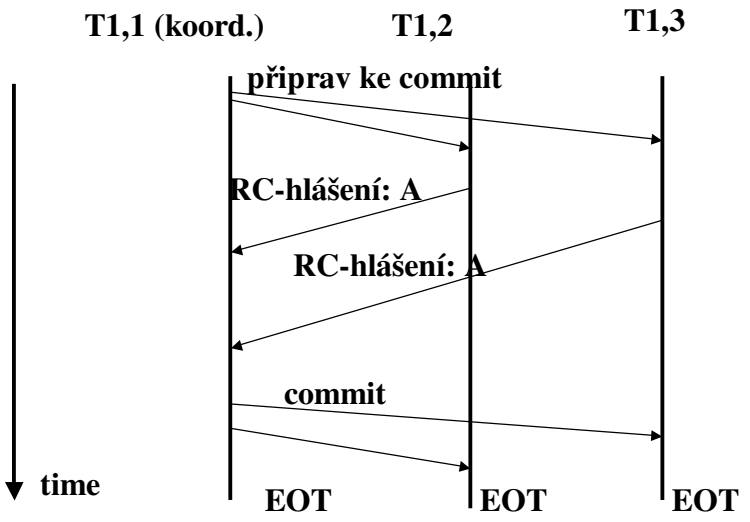


2PC: úspěšné provedení



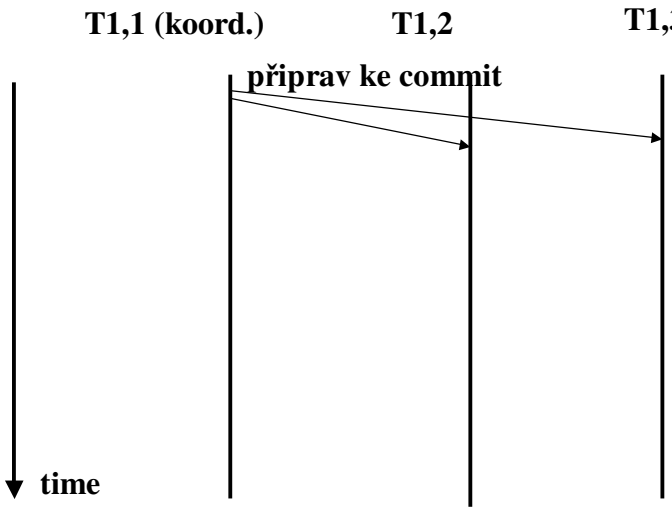
J. Pokorný - DBS2 57

2PC: úspěšné provedení



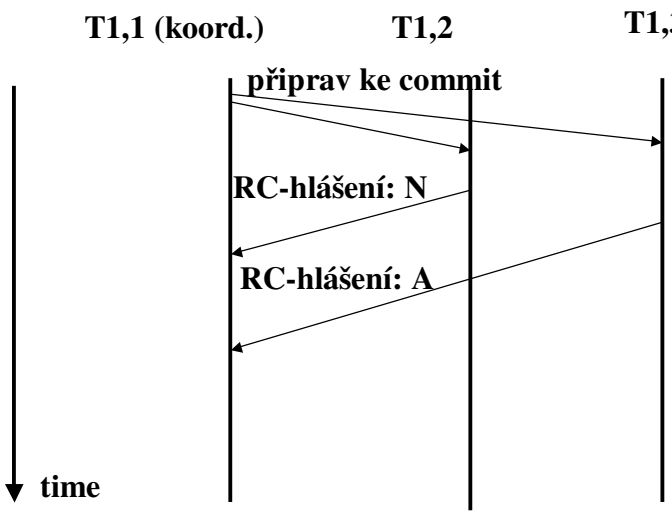
J. Pokorný - DBS2 58

2PC: neúspěšné provedení



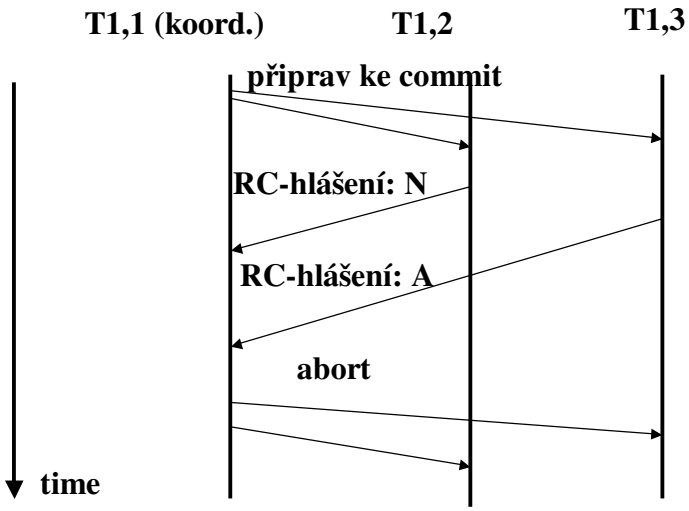
J. Pokorný - DBS2 59

2PC: neúspěšné provedení



J. Pokorný - DBS2 60

2PC: neúspěšné provedení



J. Pokorný - DBS2 61

Distribuované zotavení z chyb

- Další problémy:
  - uváznutí,
  - problémy s 2PC (co když nasatne chyb u koordinátora, co když není spojení atd.)
- Různá řešení (např. 3-phase commit)

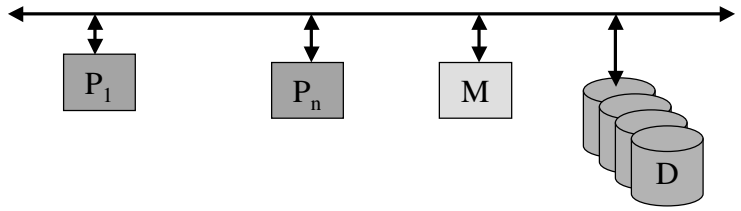
J. Pokorný - DBS2 62

Závěry

- Distribuované SŘBD nejsou příliš rozšířené
- ALE: nové myšlenky
  - polospojení
  - distribuované zotavení z chyb a souběžné zpracování
- aplikace:
  - paralelní db / klastry
  - aktivní disky
  - replikované db (servery pro e-komerci)

J. Pokorný - DBS2 63

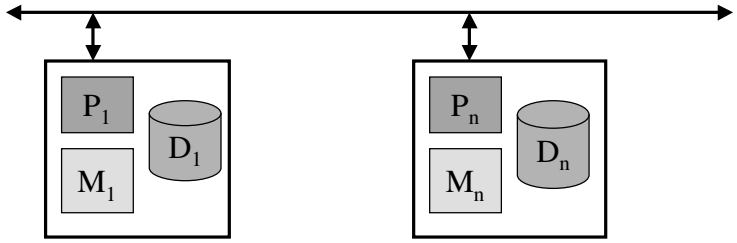
Architektura se sdílenou pamětí



Příklady: symetrické multiprocesory (Sequent, Encore) některé mainframes (IBM3090, Bull DPS8)

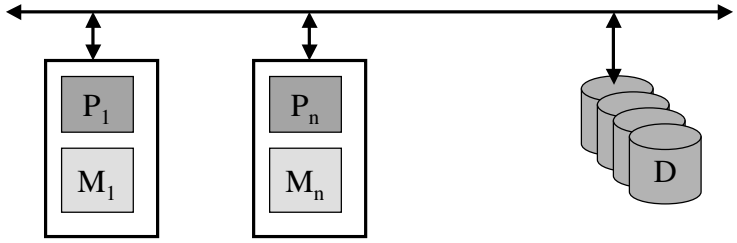
J. Pokorný - DBS2 64

Architektura s žádným sdílením



Příklady: Teradata DBC, Tandem, Intel Paragon, NCR 3600 and 3700

Architektura se sdílenými disky



Příklady: **DEC VAX Cluster, IBM IMS/VS Data Sharing**