
XML databáze

Přednáška pro kurz
PB138 Moderní značkovací jazyky
22. 4. 2003

Ing. Petr Adámek
xadamek2@fi.muni.cz
<http://www.bilysklep.cz/petr/>

XML databáze

- Proč XML databáze
 - Efektivní ukládání a vyhledávání XML dat
 - Ukládání XML v relačních databázích
 - Vybrané problémy z indexování XML dat
-
-

Proč XML databáze

- XML je univerzální formát pro výměnu informací s mnoha výhodami
 - Snadná transformace do jiných formátů
 - Ideální pro ukládání dokumentů
 - Univerzální formát pro datové modelování
 - Snazší (přirozenější) reprezentace objektů (agregace, dědičnost, relace) – viz XML Schema
-

Kdy nepoužít XML databáze

- Tam kde stačí databáze relační
- Aplikace se specifickými požadavky (výkon)
- XML databáze jsou zatím v plenkách
 - Nedosahují robustnosti databází relačních (transakce, souběžný přístup, škálovatelnost)
 - Chybí plná podpora standardů
 - Metody pro indexování a optimalizaci se zatím vyvíjí

Efektivní ukládání a vyhledávání

- XML dokumenty je nutné efektivně ukládat
- Jako úložiště lze použít
 - Souborový systém (či jiné perzistentní úložiště)
 - Relační databázi
 - Nativní XML databázi
- Indexy pro efektivní vyhledávání lze aplikovat pro všechny tři varianty

XML a relační databáze

- Pro ukládání XML lze použít relační databáze
 - Specializovaná schémata pro konkrétní aplikaci
 - Univerzální schémata bez indexování struktury
 - Univerzální schémata s indexováním struktury
- Při indexování XML dokumentů nalézá uplatnění mnoho algoritmů a technik z relačních SŘBD
- Mnoho komerčních SŘBD poskytuje podporu pro XML (rozšíření SQL).

Indexování XML dat

- Indexování XML dokumentů umožňuje
 - Efektivní vyhledávání v kolekcích dokumentů
 - Efektivní provádění XML transformací
 - Efektivní aktualizaci dokumentů
 - Efektivní navigaci v rámci dokumentu
 - Nejčastěji indexujeme pro efektivní vyhodnocování XPath výrazů, příp. vzorů
-
-

Indexování pro aplikaci XPath výrazů

- Indexování textových informací
 - Hodnoty textových uzlů
 - Hodnoty atributů
 - Jména elementů a atributů
 - Indexování strukturálních vztahů (osy XPath)
 - Vyhodnocení relace je na ose/není na ose
 - Které uzly leží na dané ose
-
-

Vyhodnocování XPath dotazů

- Pro vyhodnocení XPath výrazu je nutné provést
 - Vyhodnocení všech predikátů
 - Vyhodnocení všech strukturálních vztahů
 - Spojení (join) výsledků
 - Pořadí operací může výrazně ovlivnit efektivitu zpracování
-
-

XML Signatury

- Využívají sekvencí preorder/postorder
- Statická struktura, operace vkládání a odebírání uzlů je nákladná
- Pro efektivnější práci s obsahem os existují rozšířené signatury

Title:tree0.dia
Creator:Dia v0.90
CreationDate:Tue Apr 22 10:58:51 2003

pre: <a,b,c,d,e,g,f,h,o,p>
post: <d,e,c,g,b,o,p,h,f,a>

Sekvence preorder/postorder

$\text{pre}(x) < \text{pre}(v) \Rightarrow (x = \text{ancestor}(v)) \text{ or } (x = \text{preceding}(v))$

$\text{pre}(x) > \text{pre}(v) \Rightarrow (x = \text{descendant}(v)) \text{ or } (x = \text{following}(v))$

$\text{post}(x) < \text{post}(v) \Rightarrow (x = \text{descendant}(v)) \text{ or } (x = \text{preceding}(v))$

$\text{post}(x) > \text{post}(v) \Rightarrow (x = \text{ancestor}(v)) \text{ or } (x = \text{following}(v))$

- Takto mohu okamžitě určit vztah mezi dvěma libovolnými uzly
- Lze také generovat seznam uzlů na dané ose

Indexování pro použití vzorů

- Indexování textových informací (viz XPath)
- Indexování struktury vzorů (twigs)
 - Obsahuje dokument daný vzor?
 - Které uzly v dokumentu jej tvoří?
- Vzory lze transformovat na výrazy jazyka XPath a naopak (optimalizátor může vybrat vhodnější metodu vyhodnocení)

XML Signatory

- Lze je použít i pro hledání vzorů
- Pokud strom A obshuje podstrom B, pak také sekvence preorder(B) (resp. postorder(B)) je podsekvencí preorder(A) (reps. postorder (A))
- Podsekvence není totéž co podřetězec !
- př.: $\langle b,c,g \rangle$ je podsekvencí $\langle a,b,c,d,e,g,f,h,i,j \rangle$ a $\langle c,g,b \rangle$ je podsekvencí $\langle d,e,c,g,b,i,j,h,f,a \rangle$

Další problémy

- Optimalizace dotazů
 - Transformace výsledků do požadovaného tvaru
 - Aktualizace dokumentů
 - Podobnostní hledání
 - Předávání výsledků dotazu
 - Vyhledávání v kolekcích dokumentů
-
-

Optimalizace XPath výrazů

- Zjednodušování a transformace dotazů
 - S využitím znalosti struktury dokumentu (DTD, ...)
 - S využitím znalosti složitosti jednotlivých operací
 - S využitím statistických informací
 - Eliminace zbytečných predikátů
- Výběr vhodného pořadí pro vyhodnocování
- Výběr vhodné metody pro vyhodnocení

Závěr

- XML databáze jsou užitečný a perspektivní nástroj; nejsou však vhodné vždy
 - XML dokumenty lze ukládat různým způsobem, lze použít i relační databáze
 - Klíčovým problémem XML databázi je efektivní indexování strukturálních vztahů
-
-