Uvod u XML i XML baze podataka

Jelena Graovac
Matematički fakultet, Univerzitet u Beogradu
jgraovac@matf.bg.ac.rs
www.matf.bg.ac.rs/~jgraovac

Značaj podataka

- Podaci su često loše organizovani, teško je upravljati njima i uglavnom su nedovoljno iskorišćeni.
- Dobro organizovani podaci nisu dovoljno cenjeni s obzirom na njihov značaj.
- Rast količine podataka.
- Snažniji računari.
- Očekivanje: dobijanje kvalitetnijih informacija.
- Kvalitetnije informacije – prednost u odnosu na konkurenciju.

Vrste podataka

- Prema svojoj strukturiranosti razlikuju se:
  - Dobro strukturirani podaci
    - Na primer, telefonski imenik sa unapred jasno definisanim poljima - ime, prezime, broj telefona
    - Relacione baze podataka su podesne za skladištenje i upravljanje ovakvim podacima
  - Polustrukturirani podaci
    - Relacione baze podataka nisu najpodesnije

Šta su polustrukturirani podaci?

- Nemaju zajedničku strukturu.
- Mogu sadržati polja koja nisu poznata pre trenutka projektovanja dokumenta.
- Iste vrste podataka se mogu predstaviti na različite načine.

Predstavljanje polustrukturiranih podataka

- XML je jezik za predstavljanje polustrukturiranih podataka.
- Skraćenica od eXtensible Markup Language.
- Jezik označavanja sličan HTML-u.
- Razvijen je od strane W3C (World Wide Web Consortium) sa ciljem prevazilaženja nedostataka HTML-a.
- XML nije zamena za HTML. Tendencija je da se HTML koristi za prikazivanje a XML za opisivanje podataka.

Prednosti XML-a

- Sintaksa etiketa nije fiksirana.
- Ne mora se definisati shema.
- Fleksibilan je i proširiv, omogućava postojanje različitih tipova podataka u okviru jednog dokumenta.
- Opisuje podatke stavljanjem akcenta na to šta podaci jesu a ne kako oni izgledaju.
- Ima format koji je čitljiv za čoveka.
- Ima mogućnost internacionalnog korišćenja zahvaljujući činjenici da koristi Unicode kodnu šemu.
- Nezavisan je od platforme odnosno od softvera i hardvera koji se koristi.
- Postoji veliki broj gotovih aplikacija za procesiranje XML-a koje se mogu koristiti.

Nedostaci XML-a

- Ne mora se definisati shema.
- Fleksibilan je i proširiv.
- Manipulisanje podacima često sporije.
- Optimizacija je kompleksnija zahvaljujući bogatstvu i velikoj izražajnoj moći upitnih jezika koje koristi.

Struktura XML dokumenta

- XML dokument je samoopisjujuća, platformski nezavisna tekstualna datoteka
  - Sastoji se iz
    - Podataka (teksta)
    - Etiketa

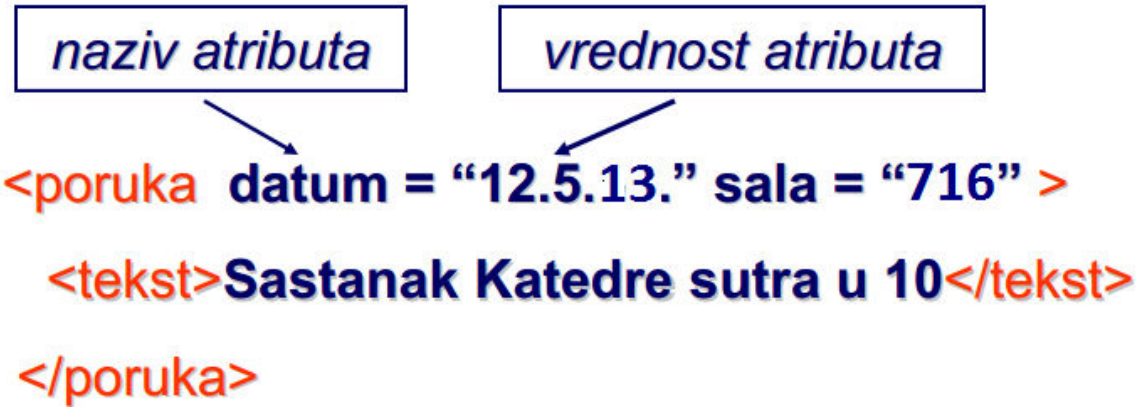
Struktura XML dokumenta

- Elementi su osnovni blokovi XML-a
  - Kontejner element – par etiketa (početna i krajnja etiketa sa sadržajem)

    <pozdrav> Hello XML! </pozdrav>
  - Prazan element – obično se za krajnju etiketu koristi skraćenica />
  - <poruka/>
  - <pozdrav tekst= “Hello XML”/>

Struktura XML dokumenta

- Elementima se mogu pridružiti atributi
  - Obezbeđuju dodatne informacije o elementima



- Imena XML etiketa i imena atributa – case sensitive

Struktura XML dokumenta

- Hijerarhijska struktura (stablo) koja se sastoji iz elemenata, atributa i znakovnih podataka
- XML dokument ima jedan i samo jedan koreni (root) element
- Svi ostali elementi u strukturi su elementi “deca“ korenog elementa
  - Dozvoljeno je višestruko ugnježdavanje elemenata

XML deklaracija

- Svaki XML dokument mora da sadrži XML deklaraciju, odnosno instrukciju obrade kojom se dokument identifikuje kao XML dokument.
  - Osnovni oblik XML deklaracije:

    <?xml version="1.0"?>
  - Opcioni oblik XML deklaracije:

    <?xml version="1.0"encoding= "UTF-8"?>

XML deklaracija

<?xml version="1.0"encoding= "UTF-8"?>

? Oznaka za instrukciju obrade

– Instrukcija obrade je poruka programima koji procesiraju XML dokument

- Atribut version specificira XML verziju
- Atribut encoding definiše znakovni kod u kome je XML dokument napisan
  - UTF-8 (kompresovana verzija Unicode-a)
  - UTF-16 (Unicode)

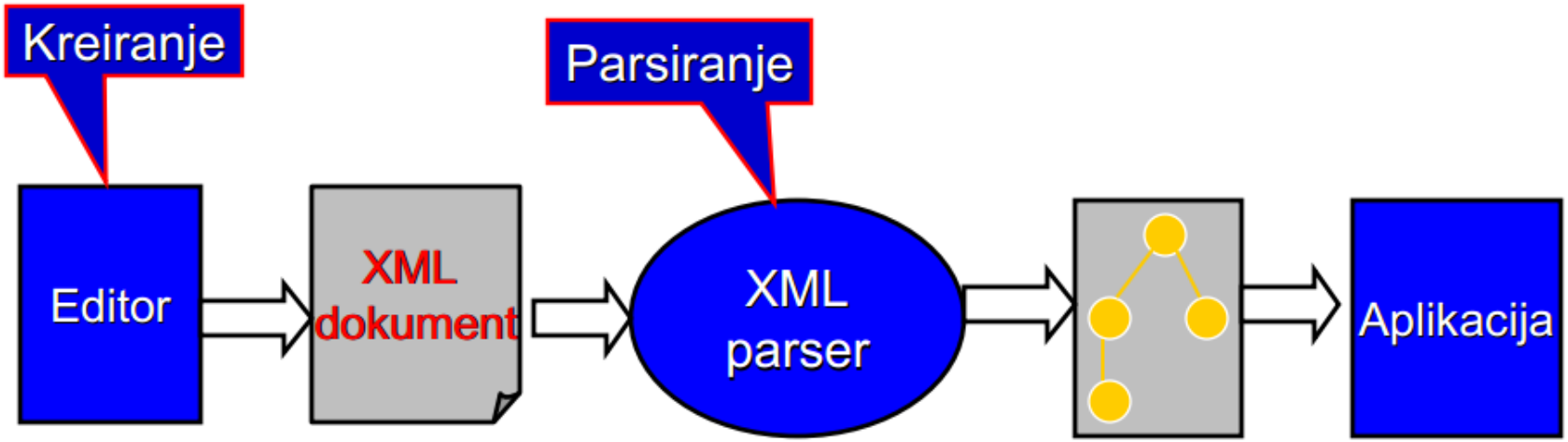
Dobro oformljen XML dokument

- Postoji XML deklaracija
- Dokument sadrži jedan i samo jedan koreni element u kome su ugnježdeni svi ostali elementi i njihovi sadržaji
- Svi elementi i atributi u dokumentu moraju da budu sintaksno ispravni

Provera sintaksne korektnosti XML dokumenta

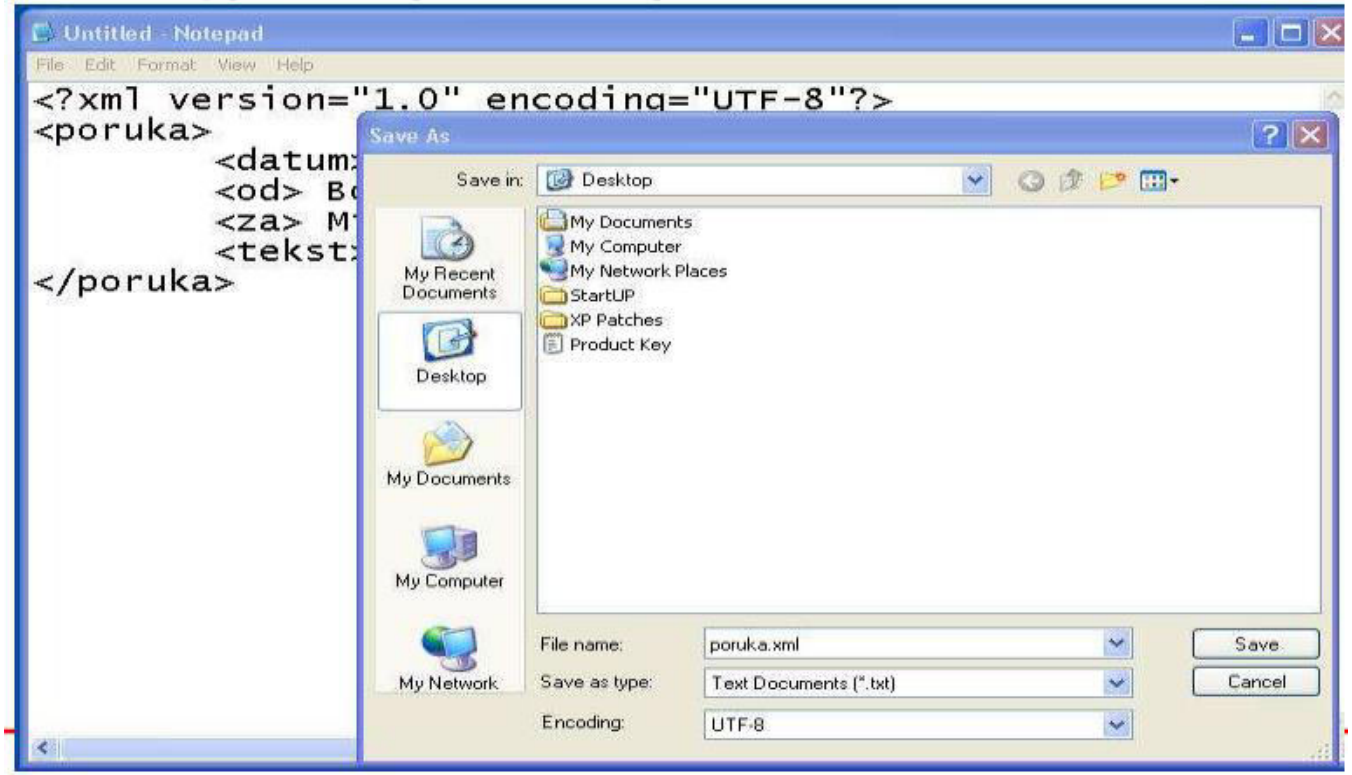
- XML parser verifikuje da li je XML dokument dobro-oformljen
- XML parser čita dokument i konvertuje ga u hijerarhijsku strukturu
- XML parser prenosi parsirani dokument do krajnje aplikacije

Obrada XML dokumenta



Kreiranje XML dokumenta

- Tekst editori (na primer Notepad)
- VS.NET XML Desinger
- XML Spy – razvojno okruženje za XML



Validacija XML dokumenta

- Definisanje tipova XML dokumenata
- W3C je ponudio dva standarda za definisanje tipova XML dokumenta, odnosno opisivanje strukture XML dokumenta:
  - Document Type Definition (DTD)
  - XML Schema Definition (XSD)
- Validni XML dokument
  - Dobro-oformljen
  - Konzistentan sa strukturom definisanom u opisu tipa dokumenta

Definisanje tipova dokumenata

- DTD i XSD definišu:
  - Strukturu XML dokumenta
  - Ime i tip svakog XML elementa/atributa
- DTD
  - Nasleđen od SGML-a
  - Poseban jezik za opis strukture dokumenta
  - Vrlo ograničene mogućnosti za definisanje tipova

Primer DTD

```
<!ELEMENT Knjige (Knjiga+)>
<!ELEMENT Knjiga (Naslov, Autor, Godina, ISBN, Izdavac)>
<!ELEMENT Naslov (#PCDATA)>
<!ELEMENT Autor (#PCDATA)>
<!ELEMENT Godina (#PCDATA)>
<!ELEMENT ISBN (#PCDATA)>
<!ELEMENT Izdavac (#PCDATA)>
```

- #PCDATA – Parser Character Data, označava znakovni sadržaj
- “+”– element se pojavljuje bar jednom

XML Schema

- Preporuka W3C od maja 2001
- Uputstvo (tutorijal) se može naći na adresi:

  http://www.w3schools.com/schema
- Data je preko XML sintakse (XML shema je XML dokument)
- Podržava definicije prostih i složenih tipova i poseduje napredne mehanizme za grupisanje XML elemenata u XML dokumentu

Schema element

- Deklaracija XML namespace:

  xmlns:prefix=“namespace name“

  – Prefix se koristi kao skraćeno ime za “namespace name“ u XML shemi
  – “namespace name“ je lokacija definicije XSD i specificira se preko URL
  – Primer

```
<?xml version="1.0"?>
<xsd:schema
  xmlns:xsd=http://www.w3.org/2001/XMLSchema>
        ……
</xsd:schema>
```

Deklaracija elementa u XML shemi

- Za svaki element u XML shemi definiše se naziv i tip (atributi name i type)

```
<xs:element name="Autor" type="xsd:string"/>
```

- Tip može da bude:
  - Korisnički definisan tip (npr. compexType)
  - U opsegu imena definicije XML sheme (npr. string)
  - Kardinalnost elementa može da se specificira u njegovom ocu-elementu. Inače, podrazumevano, kardinalnost elementa je:
  - `minOccurs="1", maxOccurs="1"`

Definisanje složenih tipova

- Složeni tipovi se konstruišu od prostih i drugih složenih tipova korišćenjem konstruktora:
  - Sequence – def. uređenu grupu elemenata koji moraju da se javljaju u definisanom redosledom. Podrazumevano, svaki element je obavezan i jednoznačan ali se to može promeniti indikatorima minOccurs i maxOccurs.
  - Choice – def. grupu iz koje se može izabrati samo jedan element
  - All – def. grupu u kojoj se svi elementi mogu pojaviti u proizvoljnom redosledu ali tačno jedanput.

Primer

```
<?xml version="1.0" encoding="UTF-8"?>
- <xsd:schema xmlns:xsd="http://www.w3.org/2000/10/XML Schema">
  - <xsd:element name="Knjige">
    - <xsd:complexType>
      - <xsd:sequence maxOccurs="unbounded">
          <xsd:element name="Knjiga" type="TipKniga"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  - <xsd:complexType name="TipKnjiga">
    - <xsd:sequence>
        <xsd:element name="Naslov" type="xsd:string"/>
        <xsd:element name="Autor" maxOccurs="unbounded" type="xsd:string"/>
        <xsd:element name="Godina" type="xsd:string"/>
        <xsd:element name="ISBN" type="xsd:string"/>
        <xsd:element name="Izdavac" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
</xsd:schema>
```

knjige.xsd

Primer

```
<?xml version="1.0" encoding="UTF-8"?>
- <Knjige xsi:noNamespaceSchemaLocation="D:\NASTAVA\pbp\XMLbaze\knjige.xsd"
  xmlns:xsi="http://www.w3.org/2000/10/XMLSchema-instnace">
    - <Knjiga>
        <Naslov>Uvod u relacione baze podataka</Naslov>
        <Autor>Gordana Pavlović-Lažetić</Autor>
        <Godina>1999</Godina>
        <ISBN>8675890117</ISBN>
        <Izdavac>Matematički fakultet, Studentski trg 16, Beograd</Izdavac>
      </Knjiga>
    - <Knjiga>
        <Naslov>Database Management Systems, 2nd Edition</Naslov>
        <Autor>Ramakrishnan Raghu</Autor>
        <Autor>Gehrke Johannes</Autor>
        <Godina>2000</Godina>
        <ISBN>0072322063</ISBN>
        <Izdavac>McGraw-Hill Higher Education, Boston, MA</Izdavac>
      </Knjiga>
  </Knjige>
```

XML dokument konstruisan u skladu sa XML shemom knjige.xsd

Da li je XML baza podataka?

- Ako se pod bazom podataka podrazumeva bilo kakva kolekcija podataka, XML u tom striktnom smislu, može se smatrati bazom podataka.
- Da li XML i tehnologije koje ga okružuju mogu pretstavljati sistem za upravljanje bazama podataka?
  - XML može obezbediti skladište podataka (XML dokumenti), sheme (DTD, XML sheme), upitne jezike (XQuery, XPath, XQL, XML-QL), interfejse.
  - Nedostaje: efikasno skladištenje, indeksi, bezbednost, transakcije i integritet podataka, pristup od strane više korisnika, upiti nad više dokumentata odjednom.

XML dokumenti

- XML dokumenti upadaju u dve široke kategorije:
  - Data-centric – dokumenti orijentisani ka podacima
  - Document-centric – dokumenti orijentisani ka sadržaju

XML dokumenti

- Data-centric – dokumenti orijentisani ka podacima
  - Karakterišu se regularnom strukturom, fino zrnastim podacima bez mešanog sadržaja.
  - Oni su projektovani tako da se koriste uglavnom za obradu od strane mašina pre nego za ljudsku upotrebu.
  - Ovakvi podaci su najčešće smešteni u nekoj relacionoj bazi podataka i javlja se potreba za transferom podataka iz relacione baze u XML dokument, iz XML dokumenta u relacionu bazu podataka ili u oba smera.

Primer data-centric dokumenta

```
<?xml version="1.0" encoding="UTF-8"?>
- <meni datum="5.10.2007">
  - <jelo>
      <ime>Domaca pileca supa</ime>
      <cena>100.00</cena>
      <kalorije>650</kalorije>
    </jelo>
  - <jelo>
      <ime>Francuski tost</ime>
      <cena>30.50 din</cena>
      <kalorije>300</kalorije>
    </jelo>
  - <jelo>
      <ime>Bakin dorucak</ime>
      <cena>200.00 din</cena>
      <kalorije>350</kalorije>
    </jelo>
</meni>
```

XML dokumenti

– Document-centric – dokumenti orijentisani ka sadržaju
  - Karakterišu se manje regularnom ili neregularnom strukturom, krupno zrnastim podacima i ima dosta mešanog sadržaja.
  - Dokumenti projektovani uglavnom za ljudsku upotrebu.
  - Redosled elemenata često nije od značaja.
  - Najčešće su ručno pisani u XML-u.

Primer document-centric dokumenta

```
<?xml version="1.0" encoding="UTF-8"?>
- <Proizvod>
    <Ime>KIRKOLINA – čaj za mršavljenje</Ime>
    <Proizvodjac>Kirka-Pharma</Proizvodjac>
  - <Opis>
    - <Paragraf>
        Predstavlja mešavinu lekovitog bilja koje kombinovanim dejstvom regulišu
        promet materija u organizmu, ubrzavaju sagorevanje masnih naslaga i utiču na
        smanjenje telesne težine.
        <i>Krušina, sena, zova</i>
        stimulišu metabolizam, podstiču probavu, smanjuju nadutost.
        <i> Breza, pirevina, rastavić </i>
        eliminišu nakupljene toksične materije, poboljšavaju cirkulaciju.
        <i>Matičnjak</i>
        oslobađa od stresa koji je često uzrok nekontrolisanog konzumiranja hrane.
        <i>Žalfija</i>
        kao izuzetni antiseptik štiti od mogućih infekcija i  utiče na jačanje organizma.
      </Paragraf>
    - <Paragraf>
        <b>Možete:</b>
      </Paragraf>
    - <List>
      - <Item>
          <Link URL="Naruci.html">Naručiti svoj čaj za mršavljenje</Link>
        </Item>
      - <Item>
          <Link URL="Kirkolina.htm">Pročitati više o ovom proizvodu</Link>
        </Item>
      - <Item>
          <Link URL="Katalog.zip">Skinuti katalog naših proizvoda</Link>
        </Item>
      </List>
  </Opis>
</Proizvod>
```

Vrste XML baza podataka

- XML-proširene baze podataka.
  - Koristi postojeći sistem za upravljanje bazama podataka.
  - Preslikava XML podatke u sopstveni model.
    - Čuvaju se hijararhija i podaci.
    - Gubi se identitet dokumenta, redosled čvorova na istom nivou,...
  - Data-centric dokumenta
- Izvorne XML baze podataka.
  - Baze podataka koje smeštaju XML u “izvornom” obliku održavajući prirodnu drvoliku strukturu ovih dokumenata.
  - Document-centric dokumenta.

Šta su izvorne XML baze podataka?

- Osnovne osobine izvornih XML baza podataka:
  - XML dokument je osnovna logička jedinica, kao što je to vrsta u tabeli kod relacionih baza.
  - Minimalno, model mora uključiti elemente, atribute, tekstualne podatke (PCDATA) i redosled dokumenta.
  - Nema zahteva za postojanjem bilo kakvog specifičnog fizičkog modela skladištenja.

Osobine izvornih XML baza podataka

- Kolekcije dokumenata.
- Ažuriranje i brisanje baze.
- Transakcije, zaključavanje i konkurencija.
- Programski interfejsi (Application Programming Interfaces — APIs). Najpoznatiji su XML:DB API i XQJ (XQuery API for Java).
- Kružno putovanje (Round-Tripping).
- Indeksi.
  - Vrednosni indeksi - “Pronaći sve elemente čija je vrednost Santa Cruz”
  - Strukturalni indeksi -“Pronaći sve City elemente čija je vrednost Santa Cruz”
  - Full-text indeksi - “Pronaći sva dokumenta koja sadrže reč Santa Cruz”

Kako izabrati najbolje rešenje

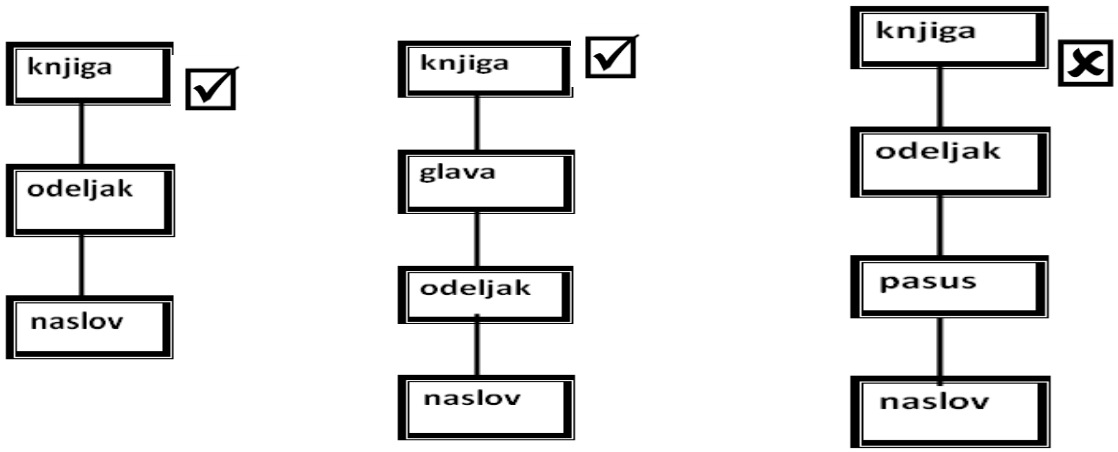
- Pri izboru baze podataka prvo pitanje na koje treba dati odgovor jeste ”Zbog čega postoji potreba za korišćenjem baze podataka i na kakav način se ta baza želi koristiti?”
- Možda najznačajniji faktor u izboru baze podataka je da li će se u nju smeštati podaci ili dokumenti.
- Ako se podacima pristupa u skladu sa predefinisanom hijerarhijom ili ako nije definisana shema dokumenta, izvorne baze podataka imaju prednost.
- Ako se očekuju česta ažuriranja onda izvorne baze nisu najbolje rešenje.
- Neke upite je lakše postaviti nad izvornim XML bazama nego nad relacionim bazama.

Jezici za postavljanje upita

- Veliki broj jezika je kreiran za postavljanje upita nad XML dokumentima a najznačajniji su Xpath i XQuery.
- XPath je W3C preporuka a sa pojavom XQuery postaje još popularniji. Koriste se za dobijanje i manipulisanje podacima iz XML baza podataka.
- Veliki broj ugrađenih funkcija.
- Korisnik ima mogućnost definisanja sopstvenih funkcija.
- Indeksi su potrebni radi efikasnijeg izvršavanja upita nad velikim kolekcijama dokumenata.

XPath

- Xpath koristi iskaze putanja za kretanje kroz logičku, hijararhijsku strukturu XML dokumenta.
- Dizajniran je da radi sa jednim XML dokumentom. Vrednost vraćena XML upitom je skup čvorova.
- Primer: Za sve putanje koji počinju od elementa `knjiga`, ispitati da li se u okviru njih nalazi element `odeljak` i vratiti kao rezultat element `naslov` koji je dete elementa `odeljak`.
  - `knjiga//odeljak/naslov`



XPath - primeri

- Selektovati sve elemente starost u dokumentu.

  `//starost`
- Selektovati sve elemente koji su deca korenog elementa student

  `/student/*`
- Selektovati sve studbr atribute elemenata student u dokumentu.

  `/student[@studbr]`
- Selektovati sve elemente starost.

  `//*[name()='starost']`
- Selektovati sve pretke od svih elemenata starost koji su deca od elementa student.

  `/student/starost/ancestor::*`

XQuery — XML Query Language

- XQuery je jezik koji je projektovan da bude mali, da se lako implementira i da bude lako razumljiv jezik.
- On je nastao sa idejom da obezbedi upitni jezik koji ima istu širinu funkcionalnosti kao SQL nad relacionim bazama podataka.
- Izrazi u XQuery-u upadaju u 6 širokih tipova:
  - Izrazi putanje.
  - Konstruktori elemenata.
  - FLWR izrazi.
  - Uslovni izrazi.
  - Kvantifikovani izrazi.
  - Izrazi koji u sebi uključuju korisnički definisane funkcije.

XQuery – izrazi putanje

- XQuery obezbeđuje izraze putanja koje su nadskup od onih u XPath-u.
  - Iz dokumenta koji sadrži zapslene i njihovu mesečnu zaradu, izdvojiti godišnju zaradu za zaposlenog sa imenom Marko.

    ```
    //zaposleni[ime="Marko"]/zarada * 12
    ```
  - U dokumentu ”zoo.xml” pronaći sve slike u poglavljima od 2 do 5

    ```
    document("zoo.xml")//poglavlje[2 TO 5]//slika
    ```

XQuery – konstruktori elemenata

- Ponekad je neophodno za upit da kreira ili generiše elemente. Takvi elementi se mogu generisati direktno u upitu u okviru iskaza nazvanog konstruktori elemenata.
  - Generisati elemente <zaposleni> koji imaju `zapid` atribute. Vrednost atributa i sadržaj elementa su specifikovani promenljivom $id koja je dodeljena u nekom drugom delu upita.

```
<zaposleni zapid = {$id}>
    {$ime}
    {$posao}
</zaposleni>
```

Xquery – FLWR iskazi

- FLWR se izgovara kao ”flower”.
- Ovaj izraz je upit koji se sastoji od FOR, LET, WHERE I RETURN klauze.
  - Izlistati sve izdavače koji su izdali više od 100 knjiga.

```
<veliki_izdavaci>
  { FOR $p IN distinct(document("bib.xml")//izdavac)
    LET $b := document("bib.xml")//knjiga[izdavac = $p]
    WHERE count($b) > 100
    RETURN $p
   }
   </ veliki_izdavaci >
```

Xquery – uslovni izrazi

- Uslovni izrazi ocenjuju test izraze i onda vraćaju jedan od dva rezultujuća izraza. Ako je vrednost test izraza tačno onda se vraća kao rezultat vrednost prvog rezultujućeg izraza, u suprotnom, vraća se vrednost drugog.
  - Napraviti listu svih knjiga uređenih po naslovu. Za beletristiku, uključiti izdavača a za sve ostale autora.

```
FOR $k IN //knjiga  RETURN
    <knjiga>
    {$k/naslov, IF ($k[@zanr = "Beletristika"]) THEN
$k/izdavac ELSE $k/autor}
    </knjiga>
     SORTBY (naslov)
```

Xquery – kvantifikovani izrazi

- SOME klauza i EVERY klauza - ekvivalentne kvantivikatorima koji se koriste u matematici i logici.
  - Pronalaći naslove svih knjiga u kojima su ”jedrenje” i ”surfovanje” pomenuti u nekom paragrafu.

```
FOR $k IN //knjiga
WHERE SOME $p IN $b//paragraf SATISFIES
(contains($p, "jedrenje") AND contains($p,
"surfovanje"))
RETURN $k/naslov
```

  - Pronalaći naslove knjiga u kojima se ”jedrenje” pominje u svakom paragrafu.

```
FOR $k IN //knjiga
WHERE EVERY $p IN $k//paragraf SATISFIES
contains($p, "jedrenje")
RETURN $k/naslov
```

Xquery – izrazi koji u sebi uključuju korisnički definisane funkcije

- Osim toga što je podržana centralna biblioteka funkcija sličnih onima u XPath-u, XQuery takođe daje mogućnost korisnicima da definišu funkcije koje će proširiti ovu biblioteku.
- Spisak korisnički definisanih funkcija može se naći na adresi:http://www.w3schools.com/xpath/xpath_functions.asp
- Primer: Napisati funkciju koja za knjigu za koju su date informacije o ceni i popustu (u procentima), izračunati cenu sa popustom.

```
declare function local:minCena($cena as xs:decimal?,
$popust as xs:decimal?) AS xs:decimal?
{let $pop := ($cena * $popust) div 100
return ($cena - $pop)};
(: Primer poziva ove funkcija je::)
<minCena>{local:minCena($knjiga/cena,$knjiga/popust)}
</minCena>
```

Izvorne XML baze podataka

- Najpoznatiji sistemi za upravljanje izvornim XML bazama podataka su:
  - eXist
    - Open source sistem za upravljanje izvornim XML bazama podataka, koji jednostavno može biti integrisan u druge aplikacije koje koriste i obrađuju XML.
    - Baza podataka je potpuno napisana u Javi.
    - http://exist-db.org
  - Berkeley DB XML
  - Oracle XML DB
  - MarkLogic Server, izvorna XML baza podataka koja koristi XQuery.

Literatura

- Bourett, Ronald. XML and Databases. http://www.rpbourret.com/xml/XMLAndDatabases.htm
- Dare Obasanjo. An Exploration Of XML In Database Management Systems. http://www.uni-weimar.de/~bauinf/lehre/Bi_3/Vorlesung/Obasanjo_XMLandDatabases.pdf
- Gordana Pavlović-Lažetić. Native XML databases vs. relational databases in dealing with xml documents.
- Dr. Milica Vučković. Uvod u XML i XML tehnologije.
- Jelena Graovac. XML baze podataka u upravljanju leksičkim resursima. Magistarska teza. http://poincare.matf.bg.ac.rs/~jgraovac/publications/jt.pdf