

XML imenski prostori

Gagulić, Kristina

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Humanities and Social Sciences / Sveučilište Josipa Jurja Strossmayera u Osijeku, Filozofski fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:142:056505>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-03-05**



FILOZOFSKI FAKULTET
SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

Repository / Repozitorij:

[FFOS-repository - Repository of the Faculty of Humanities and Social Sciences Osijek](#)



Sveučilište J. J. Strossmayera u Osijeku

Filozofski fakultet Osijek

Prijediplomski studij Informatologije

Kristina Gagulić

XML imenski prostori

Završni rad

Mentor: prof. dr. sc. Boris Bosančić

Osijek, kolovoz 2024.

Sveučilište J. J. Strossmayera u Osijeku

Filozofski fakultet Osijek

Prijediplomski studij Informatologije

Kristina Gagulić

XML imenski prostori

Završni rad

Društvene znanosti, informacijske i komunikacijske znanosti, informacijski
sustavi i informatologija

Mentor: prof. dr. sc. Boris Bosančić

Osijek, kolovoz 2024.

Prilog: Izjava o akademskoj čestitosti i o suglasnosti za javno objavljivanje

Obveza je studenta da donju Izjavu vlastoručno potpiše i umetne kao treću stranicu završnoga, odnosno diplomskog rada.

IZJAVA

Izjavljujem s punom materijalnom i moralnom odgovornošću da sam ovaj rad samostalno napisao/napisala te da u njemu nema kopiranih ili prepisanih dijelova teksta tuđih radova, a da nisu označeni kao citati s navođenjem izvora odakle su preneseni.

Svojim vlastoručnim potpisom potvrđujem da sam suglasan/suglasna da Filozofski fakultet u Osijeku trajno pohrani i javno objavi ovaj moj rad u internetskoj bazi završnih i diplomskih radova knjižnice Filozofskog fakulteta u Osijeku, knjižnice Sveučilišta Josipa Jurja Strossmayera u Osijeku i Nacionalne i sveučilišne knjižnice u Zagrebu.

U Osijeku 05.09.2024.

Kristina Gagulić, 0122237005

Ime i prezime studenta, JMBAG

Sadržaj

| | |
|---|----|
| 1. UVOD | 1 |
| 2. OPĆENITO O XML-U | 3 |
| 2.1. Povijest i osnove XML-a | 3 |
| 2.2. Sintaksa i struktura XML-a | 4 |
| 2.3. XML i DTD | 5 |
| 3. XML IMENSKI PROSTORI: KLJUČ ZA BOLJE UPRAVLJANJE XML ELEMENTIMA | 7 |
| 3.1. Općenito o imenskim prostorima | 7 |
| 3.2. XML Schema i XML imenski prostori | 7 |
| 3.3. Aplikacijski profili i XML imenski prostori | 11 |
| 4. PRIMJER PRIMJENE XML IMENSKIH PROSTORA | 13 |
| 5. ZAKLJUČAK | 18 |
| POPIS LITERATURE | 19 |

SAŽETAK

Svrha ovog rada je prikazati i pojasniti koncept XML imenskih prostora te njihovu primjenu u računalnom okruženju. Sukladno svrsi rada, ciljevi su: prikazati povijest, sintaksu i strukturu XML-a; objasniti što su XML imenski prostori, prikazujući povijest njihova nastanka i razvoja te njihovu sintaksu; objasniti što je XML Schema; dati osvrt na aplikacijski profil; i naposljetku, prikazati primjere korištenja XML imenskih prostora i aplikacijskog profila. Rad detaljno prikazuje razloge nastanka XML imenskih prostora te ističe prednosti njihova korištenja. Opisuje što je XML Schema, prikazuje njezinu sintaksu i definira aplikacijski profil, objašnjavajući njegove funkcije i način kreiranja. Na primjeru kataložnog zapisa prikazuje se kako se XML imenski prostori mogu koristiti za označavanje sadržaja te kako se sadržaj može opisati koristeći DC, VRA Core i MODS sheme metapodataka. Na kraju rada istaknute su prednosti korištenja XML imenskih prostora te njihova važnost u svrhu postizanja interoperabilnosti.

Ključne riječi: XML, imenski prostori, XML imenski prostori, XML Schema, aplikacijski profil.

1. UVOD

Kada je u ožujku 1989. godine Tim Berners-Lee, poznat kao „otac weba“, iznio prijedlog, odnosno ideju za ono što je danas poznato kao *World Wide Web* (WWW), opisao ju je kao povezani informacijski sustav. Daljnjim razmatranjem te ideje, Berners-Lee je zaključio da taj informacijski sustav omogućava prikupljanje informacija na način da sustav raste i razvija se usporedno s rastom i razvojem podataka koje sadrži. Berners-Lee također je umjesto tradicionalnog hijerarhijskog sustava razmatrao mrežu (*web*) kao vezu (*link*) u kojoj, metaforički rečeno, veza unutar stabla može postojati između listova različitih grana stabla. Danas je poznato kako *World Wide Web* obuhvaća milijune dokumenata koji su pisani HTML (*HyperText Markup Language*) jezikom i koji su međusobno povezani hipervezama.¹ Internet i WWW razvijeni su prema općeprihvaćenim standardima kojima se omogućuje da različiti tipovi računala koji koriste različite aplikacije i operacijske sustave, mogu zajedno komunicirati, a za te je standarde zadužen *World Wide Web Consortium* (W3C).² Osim standarada koji su potrebni kako bi računala mogla međusobno komunicirati, označavanje sadržaja, odnosno označiteljski jezici uvelike pomažu strojevima da „razumiju“ sadržaj dokumenta. Označiteljski jezici služe za to da se sadržaj i struktura odvoje od prikaza te da se sadržaj što bolje opiše i označi.³ Jedan od prvih pokušaja označiteljskog jezika bio je GML (*Generalized Markup Language*) kojega su izumili C. Goldfarb, E. Mosher i R. Lorie, a 1980. godine GML postaje standard (ISO 8879).⁴ „Praotac“ svih jezika za označavanje je SGML (*Standard Generalized Markup Language*).⁵ Najpoznatija aplikacija SGML-a je već spomenuti HTML, osnovni jezik za izradu mrežnih stranica.⁶ XML (*Extensible Markup Language*) je označiteljski jezik koji je naslijedio SGML. Članovi radnih grupa W3C-a shvatili su da HTML ne pruža mogućnost dovoljnog proširenja da bi na odgovarajući način obradio svaki tip podatka, a da pritom ne postane preglomazan. Umjesto da izrađuju čitav niz označiteljskih jezika za različite primjene, izrađen je XML, proširivi označiteljski jezik koji, pomoću DTD-ja (*Document Type Definition*) omogućuje prikaz sadržaja koji se ne može na

¹ Usp. Pitts-Moultis, Natanya. Osnove XML: brz i lak pristup ključnim vladajućim veštinama i tehnologijama. Čačak: Kompjuter biblioteka, 2000. Str. 31.

² Usp. Isto, str. 4.

³ Usp. Isto, str. 22.

⁴ Usp. Isto, str. 14.

⁵ Usp. Isto, str. 7.

⁶ Usp. Kirasić, Damir. XML tehnologija i primjena u sustavima procesne informatike. // Proceedings of the 28th International Convention MIPRO 2005. Rijeka: Hrvatska udruga za informacijsku i komunikacijsku tehnologiju, elektroniku i mikroelektroniku - MIPRO, 2005 ; str. 79. URL: <https://www.bib.irb.hr:8443/198358> (2024-08-24)

odgovarajući način označiti i prikazati HTML-om.⁷ Osim DTD-ja, za opisivanje sadržaja i razlikovanje elemenata u XML-u koriste se XML Schema i XML imenski prostori.⁸

Svrha ovog rada je pobliže pojasniti koncept XML imenskih prostora te njihovu primjenu u računalnom okruženju. Ciljevi rada su: prikazati povijest, strukturu i sintaksu XML-a i XML imenskih prostora, pojasniti što je XML Schema te dati osvrt na aplikacijski profil. U skladu sa svrhom i ciljevima rada, nakon Uvoda, u drugom poglavlju rada definira se što je XML-a, iznosi se kratak pregled povijesti nastanka XML-a te se pojašnjava sintaksa XML dokumenta. U ovom poglavlju ukratko je, na primjeru, prikazana sintaksa DTD-ja te su istaknuti nedostaci DTD-ja koji su potaknuli nastanak XML Scheme. U trećem poglavlju definira se što je XML Schema te se na primjeru prikazuje njezina sintaksa. U ovom se poglavlju iznose prednosti XML Scheme u odnosu na DTD te se objašnjavaju XML imenski prostori, razlozi njihova nastanka, prednosti koje donose te njihova sintaksa. Također je definirano što je aplikacijski profil, kako se kreira i što omogućava. Naposljetku, u četvrtom poglavlju na primjeru kataložnog zapisa prikazani su primjeri korištenja XML imenskih prostora te kreiranja aplikacijskog profila koristeći DC, VRA Core i MODS sheme metapodataka.

⁷ Usp. Pitts-Moultis, Natalya. Nav. dj., str. 7.

⁸ Usp. Kirasić, Damir. Nav. dj., str. 83.

2. OPĆENITO O XML-U

2.1. Povijest i osnove XML-a

XML se počeo razvijati 1990-ih godina zato što HTML, tada važeći standard, nije omogućavao definiranje novih elemenata potrebnih za opis sadržaja, odnosno zato što HTML nije bio proširiv jezik.⁹ U veljači 1998. godine W3C objavljuje XML kao preporuku te XML postaje prihvaćen unutar zajednice programera.¹⁰ XML je prvobitno razvijen za potrebe izdavaštva, no kasnije se počeo koristiti u svrhu efikasnije i lakše razmjene podataka na mreži.¹¹

Jedna od mogućih definicija XML-a jest da je to proširiv jezik za označavanje. Njime se zapisuju dokumenti i podatci u tekstualnom formatu. XML dokument najčešće se pohranjuje kao tekstualna datoteka koja je sastavljena od oznaka i sadržaja. XML format prvenstveno je namijenjen za strojnu obradu, no razumljiv je i ljudima. XML je potpuno neovisan o operacijskom sustavu, odnosno XML specifikacija je javno dostupna te za korištenje XML-a nisu potrebne nikakve licence.¹²

XML se također može definirati i kao metajezik za kreiranje označiteljskih jezika specifično orijentiranih prema određenoj vrsti sadržaja. XML omogućuje da se pomoću DTD-ja deklariraju elementi i atributi koji se mogu koristiti za opisivanje specifičnih vrsta sadržaja. Jedan od primjera za kreiranje označiteljskog jezika orijentiranog prema određenoj vrsti sadržaja jest CML (*Chemical Markup Language*). CML je izrađen prema XML specifikaciji, no on se koristi za opisivanje kemijskih formula, a elementi koji se u njemu mogu koristiti također su definirani DTD-jem.¹³ Osim sadržaja, XML se može koristiti i za označavanje metapodataka, odnosno informacija o drugim informacijama, a primjer za to može biti zapis o nekoj knjizi na kataložnom listiću; zapis na kataložnom listiću pruža informacije o nekom drugom izvoru informacija, odnosno u ovom slučaju o knjizi. Smatra se da bi se dosljednim korištenjem XML-a za označavanje metapodataka moglo doći do bolje organiziranih izvora na mreži što bi samim tim omogućilo lakše i brže pretraživanje.¹⁴ Iako je u XML-u moguće kreirati vlastite oznake, postoje sintaktička pravila koja svaki XML dokument mora slijediti.¹⁵ Ta su

⁹ Usp. XML. // Encyclopaedia Britannica Online. Encyclopedia Britannica, 2024. URL: <https://www.britannica.com/technology/XML> (2024-08-24)

¹⁰ Usp. Roy, Jaideep; Ramanujan, Anupama. XML schema language: taking XML to the next level. // IT Professional 2, 3(2001) ; str. 37. URL: <https://doi.org/10.1109/6294.918217> (2024-08-24)

¹¹ Usp. Pitts-Moultis, Natalya. Nav. dj., str. 5.

¹² Usp. Kirasić, Damir. Nav. dj., str. 79.

¹³ Usp. Pitts-Moultis, Natalya. Nav. dj., str. 5-6.

¹⁴ Usp. Isto, str. 6.

¹⁵ Usp. Isto, str. 90.

pravila propisana u XML specifikaciji koju koordinira W3C koji i definira XML i prateće tehnologije.¹⁶ XML specifikacija obuhvaća detaljan opis strukture XML-a te se smatra najboljim izvorom točnih informacija o XML-u.¹⁷

Neki od primjera u kojima se XML koristi za označavanje sadržaja jesu: XHTML, odnosno inačica HTML-a koja slijedi XML preporuke, jezik za definiranje matematičkih formula, odnosno MathML, razmjena poslovnih dokumenata pomoću ebXML-a i slično, a osim toga svatko tko poznaje XML može kreirati vlastiti označiteljski jezik za specifičnu primjenu.¹⁸

2.2. Sintaksa i struktura XML-a

Na najnižoj razini definiranja sastavnih dijelova XML-a, može se reći da se XML dokument sastoji od niza Unicode znakova, odnosno u XML dokumentu mogu se pojaviti svi grafemi najvažnijih svjetskih pisama, uključujući i diakritičke znakove u hrvatskom jeziku.¹⁹ Ovim se znakovima tvore dijelovi XML dokumenata od kojih su najčešći: elementi, podelementi i atributi, no još se pojavljuju i XML deklaracija, komentari, procesorske naredbe, DTD, CDATA sekcije te reference na entitete.²⁰ Neki od elemenata prikazani su na primjeru XML dokumenta u kojem je opisana zbirka knjiga koje neka osoba posjeduje (Slika 1). Elementi počinju početnom oznakom (`<knjiga>`), a završavaju završnom oznakom (`</knjiga>`), a između oznaka nalazi se sadržaj, no element može biti i prazan te se u tom slučaju piše s kosom crticom na kraju oznake (`<ocjena broj="5"/>`). XML dokument mora imati jedan korijenski element unutar kojega se nalaze svi ostali elementi u dokumentu (`<zbirka_knjiga>`). XML deklaracija mora se nalaziti na samom početku XML dokumenta. Obavezno je navesti inačicu XML-a, dok je navođenje tipa znakova koji se koriste opcionalno (`<?xml version="1.0" encoding="UTF-8"?>`). Element može imati i podelemente, pri čemu je element, onda, element-roditelj podelementu, dok je podelement element-dijete elementu (element `<knjiga>` je element-roditelj elementima `<naslov>`, `<autor>`, `<godina_izdanja>`, `<žanr>`, `<br_str>`, `<ocjena>`). Atributi služe kako bi se pobliže opisala svojstva elementa (svaki `<ocjena>` element ima atribut naziva `broj`).²¹ Komentari u XML dokumentu služe za pisanje napomena; oni se ne tretiraju kao dio sadržaja dokumenta (`<!--treći dio trilogije Misterij porijekla-->`). DTD je opcionalan, a njegova uloga je detaljno opisati i ograničiti sadržaj XML dokumenta. Primjer povezivanja s

¹⁶ Usp. Kirasić, Damir. Nav. dj., str. 80.

¹⁷ Usp. Pitts-Moultis, Natalya. Nav. dj., str. 91.

¹⁸ Usp. Kirasić, Damir. Nav. dj., str. 88.

¹⁹ Usp. Isto, str. 80.

²⁰ Usp. Isto, str. 81.

²¹ Usp. Isto, str. 80.

eksternim DTD-om je: `<!DOCTYPE zbirka_knjiga SYSTEM „zbirka_knjiga.dtd“>`. Entiteti se mogu definirati kao par ime/vrijednost.²² U primjeru *zbirka_knjiga.xml* kao entitet odabrani su inicijali vlasnika knjiga u dokumentu (Slika 2). Reference na entitet izvode se u XML dokumentu korištenjem imena entiteta deklariranog u DTD-ju (Slika 3).

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE zbirka_knjiga SYSTEM "zbirka_knjiga.dtd">
<zbirka_knjiga>

  <knjiga>
    <naslov>Ponos i predrasude</naslov>
    <autor>Jane Austen</autor>
    <godina_izdanja>2017</godina_izdanja>
    <žanr>ljubavni roman</žanr>
    <br_str>412</br_str>
    <ocjena broj="5"/>
    <vlasnica>&ime;</vlasnica>
  </knjiga>

  <knjiga>
    <naslov>Svijet Atlantide</naslov> <!--treći dio trilogije Misterij porijekla -->
    <autor>A.G. Riddle</autor>
    <godina_izdanja>2019</godina_izdanja>
    <žanr>znanstvena fantastika</žanr>
    <br_str>328</br_str>
    <ocjena broj="4"/>
    <vlasnica>&ime;</vlasnica>
  </knjiga>
</zbirka_knjiga>
```

Slika 1. Primjer XML kôda (*zbirka_knjiga.xml*).

```
<!ENTITY ime "T. T.">
```

Slika 2. Deklariranje entiteta u *zbirka_knjiga.dtd* datoteci.

```
<vlasnica>&ime;</vlasnica>
```

Slika 3. Pozivanje na vrijednost entiteta (*zbirka_knjiga.xml*).

2.3. XML i DTD

DTD definira strukturu XML dokumenta.²³ DTD opisuje komponente koje su upotrijebljene u XML dokumentu ili drugom označiteljskom jeziku koji je nastao po pravilima XML specifikacije, odnosno DTD definira elemente, attribute i entitete koji se mogu pojaviti u određenom XML dokumentu.²⁴ XML pruža mogućnost kreiranja vlastitih oznaka i DTD-ja kako bi se što preciznije opisale informacije, odnosno sadržaj dokumenta. DTD sadržava

²² Usp. Isto, str. 81.

²³ Usp. Roy, Jaideep; Ramanujan, Anupama. Nav. dj., str. 38.

²⁴ Usp. Pitts-Moultis, Natalya. Nav. dj., str. 84.

specifičnosti jezika za označavanje koje se nalaze unutar XML dokumenta, a one se pojavljuju u oblicima: deklaracije elemenata, deklaracije atributa, modela sadržaja i deklaracije entiteta.²⁵

DTD se može pisati unutar samog XML dokumenta ili se može definirati u posebnoj datoteci.²⁶ Za primjer DTD-ja primjera *zbirka_knjiga.xml* korišten je eksterni DTD (Slika 4).

```
<!ELEMENT zbirka_knjiga (knjiga+)>
<!ELEMENT knjiga (naslov,autor,godina_izdanja,žanr,br_str,ocjena,vlasnica)>
<!ELEMENT naslov (#PCDATA)>
<!ELEMENT autor (#PCDATA)>
<!ELEMENT godina_izdanja (#PCDATA)>
<!ELEMENT žanr (#PCDATA)>
<!ELEMENT br_str (#PCDATA)>
<!ELEMENT ocjena (#PCDATA)>
<!ELEMENT vlasnica (#PCDATA)>
<!ATTLIST ocjena broj (1|2|3|4|5) #REQUIRED>
<!ENTITY ime "T. T.">
```

Slika 4. Prikaz DTD-a *zbirka_knjiga.dtd*.

Kako bi XML dokument bio validan, potrebno je sve oznake koje su korištene za opis sadržaja definirati u DTD-ju ili XML Schemi.²⁷ Ne mora svaki XML dokument biti validan, dovoljno je da je dokument dobro оформljen, odnosno da se u dokumentu poštuju sintaktička pravila XML specifikacije.²⁸ Neka od pravila za dobro оформljeni XML dokument jesu: dokument započinje XML deklaracijom, dokument sadrži samo jedan korijenski element, imena elemenata ili atributa ne sadržavaju praznine, vrijednost atributa se definira unutar navodnika itd.²⁹ Validan dokument mora koristiti elemente koji su definirani u DTD-ju, uključivati sve attribute koji su u DTD-ju definirani kao obavezni te precizno pratiti model sadržaja koji je definiran za svaki element.³⁰

Iako je DTD koristan, on ima nekoliko nedostataka, a rješenje za te nedostatke može ponuditi XML Schema.

²⁵ Usp. Pitts-Moultis, Natalya. Nav. dj., str. 95.

²⁶ Usp. Kirasić, Damir. Nav. dj., str. 81.

²⁷ Usp. Isto, str. 80.

²⁸ Usp. Usp. Pitts-Moultis, Natalya. Nav. dj., str. 113.

²⁹ Usp. Kirasić, Damir. Nav. dj., str. 82.

³⁰ Usp. Pitts-Moultis, Natalya. Nav. dj., str. 122-123.

3. XML IMENSKI PROSTORI: KLJUČ ZA BOLJE UPRAVLJANJE XML ELEMENTIMA

3.1. Općenito o imenskim prostorima

Imenski prostori jedan su od važnijih 'alata' za organizaciju kôda označiteljskih jezika. Mogu se definirati kao spremnici (engl. *container*) koji sadrže skup identifikatora. Imenski prostori pružaju mogućnost da se posebno grupiraju setovi povezanih varijabli, klasa i funkcija kako bi ih se što jednostavnije razlikovalo. Osim što se koriste za organiziranje kôda, imenski prostori sprječavaju konflikte prilikom podudaranja naziva elemenata različitog značenja (engl. *naming conflicts*) u kôdu. Kada kôdovi postanu veliki, postaje sve važnije imati razvijen sustav za praćenje svih identifikatora koji se koriste, a bez imenskih prostora u svrhu razlikovanja različitih skupova korištenih identifikatora bilo bi moguće samo pratiti nazive identifikatora. Osim što bi to bio mukotrpan i dugotrajan posao, zbog velikog broja elemenata u kôdu mogu se pojaviti ista imena elemenata što može uzrokovati neispravnost kôda (engl. *bugs*), a osim toga, zbog slabije organizacije kôd će biti teži za čitanje i razumijevanje. Postoje četiri vrste imenskih prostora: globalni imenski prostori (engl. *global namespaces*) koji sadržavaju sve identifikatore koji su definirani u kôdu, lokalni imenski prostori (engl. *local namespaces*) koji se koriste kako bi se grupirali identifikatori unutar određene klase ili funkcije, ugrađeni imenski prostori (engl. *built-in namespaces*) koji predstavljaju imenske prostore koji su definirani programskim jezikom te imenski prostori u različitim programskim jezicima.³¹ Kako bi se razlikovalo različite imenske prostore, svaki imenski prostor ima naziv, odnosno jedinstveni identifikator.³²

3.2. XML Schema i XML imenski prostori

Nastanak XML Scheme može se satirično opisati sljedećom kronologijom: prvo su se ljudi žalili na prekomplikirani mehanizam DTD-ja, posebno zbog sintakse koja se razlikuje od sintakse XML-a, zatim je W3C sastavio radnu grupu kako bi riješila problem te složenosti, a na kraju je razvijena XML Schema, koja je još kompliciranija od DTD-ja.³³ XML Schema zapravo ima istu funkciju kao DTD, odnosno XML Schema predstavlja opis kojim se definira dopuštena struktura, sadržaj te ograničenja XML dokumenta.³⁴ XML Schema definirana je

³¹ Usp. What is a Namespace. URL: <https://www.programmingcube.com/what-is-a-namespace/> (2024-08-24)

³² Usp. Kirasić, Damir. Nav. dj., str. 82.

³³ Usp. Isto, str. 83.

³⁴ Usp. XML Schema // W3C, 2000. URL: <https://www.w3.org/XML/Schema> (2024-09-08).

W3C standardima.³⁵ W3C radna skupina koja nadzire razvoj XML Scheme objavila je preporuku za XML Schemu u listopadu 2000. godine. U XML Schemi također se definiraju i tipovi podataka elemenata i atributa koji se mogu pojaviti u XML dokumentu.³⁶ XML Schema se koristi za: postavljanje uvjeta i ograničenja sadržaja i strukture XML dokumenta, određivanje pravila koja je potrebno slijediti kako bi XML dokument bio validan te za definiranje rječnika oznaka i gramatike koji se mogu pojaviti u XML dokumentu.³⁷

Iako je funkcija DTD-ja i XML Scheme ista, XML Schema pruža puno više mogućnosti od DTD-ja. Primjerice, u XML Schemi moguće je definirati da sadržaj određenog elementa mora biti broj, što u DTD-ju nije moguće.³⁸ XML Schemom moguće je definirati i druge tipove podataka koji se mogu pojaviti u sadržaju koji se opisuje, a neki od njih su: običan tekst (engl. *string*), logički tip podataka (engl. *boolean*), vrijeme (engl. *time*), datum (engl. *date*) i drugi.³⁹ XML Scheme podržavaju uporabu XML imenskih prostora što omogućava da deklarirani elementi i atributi pripadaju određenom imenskom prostoru.⁴⁰ Osim toga, XML Schema je, za razliku od DTD-ja, pisana kao XML dokument, odnosno XML Schema nema različitu sintaksu od XML dokumenta (Slika 5).⁴¹

³⁵ Usp. Kirasić, Damir. Nav. dj., str. 83.

³⁶ Usp. Roy, Jaideep; Ramanujan, Anupama. Nav. dj., str. 37.

³⁷ Usp. Kirasić, Damir. Nav. dj., str. 83.

³⁸ Usp. Isto, str. 83.

³⁹ Usp. Roy, Jaideep; Ramanujan, Anupama. Nav. dj., str. 37.

⁴⁰ Usp. Isto, str. 39.

⁴¹ Usp. XML Schema. // W3Schools. URL: https://www.w3schools.com/xml/xml_schema.asp (2024-09-08).

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://knjige/zbirka_knjiga"
  xmlns="http://www.knjige.com/zbirka_knjiga"
  elementFormDefault="qualified">
  <xs:element name="zbirka_knjiga">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="knjiga" maxOccurs="unbounded">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="naslov" type="xs:string"/>
              <xs:element name="autor" type="xs:string"/>
              <xs:element name="godina_izdanja" type="xs:gYear"/>
              <xs:element name="žanr" type="xs:string"/>
              <xs:element name="br_str" type="xs:positiveInteger"/>
              <xs:element name="ocjena">
                <xs:complexType>
                  <xs:attribute name="broj" type="xs:positiveInteger" use="required"/>
                </xs:complexType>
              </xs:element>
              <xs:element name="vlasnica" type="xs:string"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Slika 5. Prikaz XML Scheme (*zbirka_knjiga.xsd*) za dokument *zbirka_knjiga.xml*.

Za potrebe stvaranja XML Scheme koja sadrži pravila o strukturi i elementima XML dokumenta *zbirka_knjiga.xml* bilo je potrebno napraviti promjenu u XML dokumentu, odnosno XML dokument se morao povezati sa XML Shemom te se entitet *ime* morao deklarirati, u ovom slučaju kao interni DTD (Slika 6).

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE zbirkaa_knjiga [
  <!ENTITY ime "T.T.">
]>
<zbirka_knjiga
  xmlns="http://knjige/zbirka_knjiga"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://knjige/zbirka_knjiga zbirka_knjiga.xsd">

```

Slika 6. Promjene u dokumentu *zbirka_knjiga.xml*.

Početkom 21. stoljeća, W3C odlučuje da će se svakom elementu koji pripada specifikaciji XML Scheme dodati prefiks *xsd* ili *xs*, kao što je vidljivo u primjeru *zbirka_knjiga.xsd* (Slika 5). W3C također je predložio da se prefiksi općenito koriste u svrhu razlikovanja dvaju elemenata koji imaju isti naziv, a različito značenje; primjerice *<table>* kao

tablica, ali i stol. Prefiksi su u vlastitim shemama metapodataka proizvoljni.⁴² Odnosno za primjer `<table>` može se koristiti `<f:table>` (*furniture*) ili `<h:table>` (HTML tablica).⁴³ Korištenjem prefiksa za označavanje metapodataka omogućuje se razlikovanje dviju shema metapodataka korištenih u nekom XML dokumentu, no budući da danas postoji mnogo standardiziranih i nestandardiziranih shema metapodataka, može se dogoditi da dođe do uporabe istog prefiksa i istog naziva elementa, čime se tada zapravo ne rješava problem istog naziva, a različitog značenja. Rješenje nude XML imenski prostori koji omogućuju da se elementima s istim prefiksom dodijeli URI (*Uniform Resource Identifier*) koji će ih učiniti jedinstvenim i razlikovnim od drugih elemenata.⁴⁴ URI predstavlja kompaktni niz znakova za identifikaciju fizičkih ili apstraktnih izvora.⁴⁵

XML imenski prostori, prema definiciji W3C-a, pružaju jednostavnu metodu za određivanje imena elemenata i atributa korištenih u XML dokumentu povezujući ih s imenskim prostorima identificiranih URI-jem.⁴⁶ W3C je XML imenske prostore objavio kao preporuku u siječnju 1999. godine čime specifikacija XML imenskih prostora postaje prikladna za široku uporabu. XML imenski prostori jednoznačno identificiraju imena elemenata i atributa u XML dokumentu. Imenski prostori tako omogućavaju da se koristi više jezika temeljenih na XML specifikaciji, bez podudaranja naziva elemenata.⁴⁷

Već je spomenuto kako se elementi istog naziva i istog prefiksa, a različitog značenja mogu pomoću XML imenskih prostora razlikovati od drugih elemenata korištenjem URI-ja. Imenski prostor deklarira se pomoću rezerviranog atributa, a naziv atributa glasi `@xmlns`.⁴⁸ Formalna sintaksa XML imenskih prostora može se prikazati formulom:

$$\text{xmlns:namespace-prefix}=\text{"namespaceURI"}$$

u kojoj atribut `@xmlns` označava da se radi o XML imenskim prostorima, *namespace-prefix* označava mjesto rezervirano za određivanje prefiksa koji će predstavljati kraticu određenog

⁴² Usp. Bosančić, Boris. XML imenski prostori i povezivanje metapodatkovnih standarda baštinskih ustanova. // 13. seminar Arhivi, knjižnice, muzeji: mogućnosti suradnje u okruženju globalne informacijske infrastrukture : zbornik radova / uredile Mirna Willer i Sanjica Faletar Tanacković. Zagreb: Hrvatsko knjižničarsko društvo, 2010 ; str. 207.

⁴³ Usp. XML Namespaces. // W3Schools. URL: https://www.w3schools.com/xml/xml_namespaces.asp (2024-08-24).

⁴⁴ Usp. Bosančić, Boris. Nav. dj., str. 208.

⁴⁵ Usp. Berners-Lee, Tim ... [et al.]. Uniform Resource Identifier (URI): Generic Syntax, 2005., str. 1. URL: <https://www.rfc-editor.org/rfc/pdfrfc/rfc3986.txt.pdf> (2024-08-24)

⁴⁶ Usp. Namespaces in XML 1.0 (Third Edition). // W3C, 2009. URL: <https://www.w3.org/TR/xml-names/> (2024-08-24)

⁴⁷ Usp. Roy, Jaideep; Ramanujan, Anupama. Nav. dj., str. 39.

⁴⁸ Usp. Namespaces in XML 1.0 Nav. dj.

XML imenskog prostora, a "*namespaceURI*" predstavlja jedinstveni identifikator jednog XML imenskog prostora.⁴⁹ Sintaksa XML imenskih prostora prikazana je na primjeru *zbirka_knjiga.xml* (Slika 6). Prvo je u korijenskom elementu naveden atribut *@xmlns*, zatim prefiks *xsi* koji predstavlja kraticu XML imenskog prostora naziva, odnosno URI-ja <http://www.w3.org/2001/XMLSchema-instance>. U ovom primjeru, navedeni XML imenski prostor primjenjuje se kako bi pomoću navedenog URI-ja identificirao elemente s prefiksom *xsi* unutar dokumenta *zbirka_knjiga.xml*. Imena XML imenskih prostora dodjeljuju se na temelju konvencije o mrežnim adresama, a dodijeljena imena u obliku URI-ja ne moraju voditi na postojeću mrežnu stranicu.⁵⁰ U primjeru *zbirka_knjiga.xsd* (Slika 5) vidljivo je da se prvo navodi XML imenski prostor za elemente iz specifikacije XML Scheme (*xmlns:xs="http://www.w3.org/2001/XMLSchema"*), zatim se navodi identifikator elemenata koji se deklariraju u XSD dokumentu (*targetNamespace="http://knjige/zbirka_knjiga"*), potom se definira zadani imenski prostor, što omogućava da se ne navodi prefiks uz nazive elemenata (*xmlns="http://www.knjige.com/zbirka_knjiga"*), a na samom kraju deklaracije označava se da svi elementi koji se deklariraju ovom XML Schemom moraju imati pripadni XML imenski prostor (*elementFormDefault="qualified"*).⁵¹ Ako se imenski prostor definira unutar elementa u XML-u, taj se imenski prostor smatra podrazumijevanim (engl. *default*) imenskim prostorom za taj element i podelemente koje sadržava.⁵² Ako se atribut *@xmlns* s određenim prefiksom pojavljuje u korijenskom elementu, tada svi podelementi koji koriste taj prefiks upućuju na jedan XML imenski prostor.⁵³

Može se reći da se XML imenski prostori primjenjuju kako bi pomoću URI-ja identificirali elemente s istim prefiksom unutar XML dokumenta.⁵⁴

3.3. Aplikacijski profili i XML imenski prostori

Kao što je već spomenuto, cilj XML-a kao označiteljskog jezika jest da na najbolji i najučinkovitiji način opiše sadržaj. Prikazano je kako tomu pridonose metapodatkovne sheme reprezentirane XML Schemama i XML imenskim prostorima, no korištenje jedne sheme metapodataka ponekad nije dovoljno kako bi se sadržaj precizno opisao. U takvim slučajevima

⁴⁹ Usp. XML Namespaces. Nav. dj.

⁵⁰ Usp. Kirasić, Damir. Nav. dj., str. 82.

⁵¹ Usp. Bosančić, Boris. Nav. dj., str. 209.

⁵² Usp. XML Namespaces. Nav. dj.

⁵³ Usp. Isto

⁵⁴ Usp. Bosančić, Boris. Nav. dj., str. 209.

poželjno je koristiti dvije ili više shema metapodataka. Posebna shema metapodataka koja kombinira elemente postojećih shema metapodataka odnosno pripadnih imenskih prostora naziva se aplikacijski profil.⁵⁵ Te sheme metapodataka, odnosno njihovi elementi, kombiniraju se na najefikasniji i najučinkovitiji način za primjenu u lokalnoj aplikaciji.⁵⁶ Osim široko rasprostranjenih standardiziranih aplikacijskih profila, postoje i 'nestandardizirani' aplikacijski profili stvoreni u metapodatkovnoj standardnoj okolini u kojoj je sasvim prihvatljivo stvarati vlastite aplikacijske profile. XML Schema koja deklarira elemente pripadnog aplikacijskog profila naziva se kontejner (engl. *container*) XML Schema.⁵⁷ Kako bi se stvorio aplikacijski profil, potrebno je poznavati XML imenske prostore metapodatkovnih shema koje će se koristiti te putanju do dokumenta u kojem su elementi iz svake metapodatkovne sheme deklarirani.⁵⁸ Primjer XML dokumenta i kontejner XML Scheme prikazat će se u idućem poglavlju.

⁵⁵ Usp. Isto, str. 212.

⁵⁶ Usp. Heery, Rachel; Manjula Patel. Application Profiles : Mixing and Matching Metadata Schemas. // Ariadne 25, 9(2000). URL: <https://www.ariadne.ac.uk/issue/25/app-profiles/> (2024-09-08)

⁵⁷ Usp. Bosančić, Boris. Nav. dj., str. 212.

⁵⁸ Usp. Isto, str. 214.

4. PRIMJER PRIMJENE XML IMENSKIH PROSTORA

Primjena XML imenskih prostora bit će prikazana na primjeru katalognog zapisa iz kataloga knjižnice Filozofskog fakulteta Osijek. Zapis je prikazan u obliku tablice (Tablica 1) koja sadrži sve elemente katalognog zapisa, a oni će biti opisani u XML dokumentu čija će se struktura definirati korištenjem XML Schema, XML imenskih prostora i aplikacijskog profila.

Tablica 1 – Grafički prikaz primjera katalognog zapisa knjižnice Filozofskog fakulteta Osijek.

| | |
|----------------------|--|
| 101105028: | monografija |
| Autor: | Krleža, Miroslav |
| Naslov: | Povratak Filipa Latinovicza |
| Impresum: | Zagreb : Minerva, 1932. |
| Materijalni opis: | 215 str. ; 20 cm |
| Nakladnička cjelina: | Sabrana djela ; [2] |
| Napomene: | Inv. br.: 74.712hk. |
| Predmetnice: | Psihološki roman -- Hrvatska književnost Socijalni roman -- Hrvatska književnost Egzistencijalističke pjesme -- Hrvatska književnost |
| UDK: | 821.163.42-31 * 821.163.42-8 |
| Signatura: | RA886.2-8; KRL, s-2 |

Jedan od mogućih načina prikaza ovog zapisa u XML dokumentu je da naziv svih elemenata bude isti kao u katalognom zapisu (Slika 7). Jedino se prvi element razlikuje u nazivu, jer nazivi elemenata u XML dokumentima ne mogu biti brojevi. Pripadajuća XML Schema u kojoj se nalaze pravila koja definiraju strukturu ovog XML dokumenta te elemente koji se u njemu nalaze jest *primjer_zapis.xsd* (Slika 8). Sav sadržaj definiran je kao običan tekst, budući da ne postoji nijedan dio sadržaja u kojemu se koriste samo brojevi bez slova ili neki drugi tip podataka.

```

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

<katalogni_zapis
xmlns="http://katalog/zapis_krleza"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://katalog/zapis_krleza_primjer_zapis.xsd">

  <identifikator_vrste>monografija</identifikator_vrste>
  <autor>Krleža, Miroslav</autor>
  <naslov>Povratak Filipa Latinovicza</naslov>
  <impresum>Zagreb : Minerva, 1932.</impresum>
  <materijalni_opis>215 str. ; 20 cm</materijalni_opis>
  <nakladnicka_cjelina>Sabrana djela ; [2]</nakladnicka_cjelina>
  <napomene>Inv. br. : 74.712hk.</napomene>
  <predmetnice>
    Psihološki roman -- Hrvatska književnost
    Socijalni roman -- Hrvatska književnost
    Egzistencijalističke pjesme -- Hrvatska književnost
  </predmetnice>
  <udk>821.163.42-31 * 821.163.42-8</udk>
  <signatura>RA886.2-8; KRL, s-2</signatura>

</katalogni_zapis>

```

Slika 7. Prikaz kataložnog zapisa u dokumentu *primjer_zapis.xml* korištenjem proizvoljnih oznaka.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://katalog/zapis_krleza"
xmlns="http://katalog/zapis_krleza"
elementFormDefault="qualified">
<xs:element name="katalogni_zapis">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="identifikator_vrste" type="xs:string" />
      <xs:element name="autor" type="xs:string" />
      <xs:element name="naslov" type="xs:string" />
      <xs:element name="impresum" type="xs:string" />
      <xs:element name="materijalni_opis" type="xs:string" />
      <xs:element name="nakladnicka_cjelina" type="xs:string" />
      <xs:element name="napomene" type="xs:string" />
      <xs:element name="predmetnice" type="xs:string" />
      <xs:element name="udk" type="xs:string" />
      <xs:element name="signatura" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Slika 8. Prikaz XML Scheme *primjer_zapis.xsd*.

Osim korištenja vlastitih proizvoljnih elemenata u XML dokumentu, kreiranja XML Scheme koja definira te elemente te XML imenskih prostora za elemente iz deklaracije XML Scheme i XML imenskog prostora u kojem se nalaze elementi korišteni u XML dokumentu, moguće je ovaj zapis prikazati i korištenjem neke široko korištene i poznate sheme

metapodataka. U ovom primjeru to će biti Dublin Core, budući da je ona danas najkorištenija shema metapodataka u mrežnom okruženju (Slika 9).⁵⁹

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

<katalogni_zapis
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:dcterms="http://purl.org/dc/terms/">
<dc:title>Povratak Filipa Latinovicza</dc:title>
<dc:creator>Miroslav Krleža</dc:creator>
<dc:subject xsi:type="dcterms:UDC">821.163.42-31</dc:subject>
<dc:subject xsi:type="dcterms:UDC">821.163.42-8</dc:subject>
<dc:description>Psihološki roman -- Hrvatska književnost
Socijalni roman -- Hrvatska književnost
Egzistencijalističke pjesme -- Hrvatska književnost</dc:description>
<dc:publisher>Minerva</dc:publisher>
<dcterms:issued>1983.</dcterms:issued>
<dc:format>215 str. ; 20 cm</dc:format>

</katalogni_zapis>
```

Slika 9. Prikaz zapisa koristeći Dublin Core shemu.

Osim Dublin Core sheme metapodataka, postoji još nekoliko široko korištenih shema metapodataka kada su u pitanju zapisi u baštinskim ustanovama, budući da se primjer odnosi na zapis u knjižnici, a neke od njih su: EAD (*Encoding Archival Description*) koju su razvili arhivisti, VRA Core (*Visual Resources Association Core*) koju su razvili muzealci te MODS (*Metadata Object Description Schema*).⁶⁰ Moguće je pridružiti elemente jedne sheme metapodataka elementima druge sheme metapodataka, a taj se postupak zove mapiranje. Mapiranjem se elementima, njihovom značenju i sintaksi pridružuju elementi druge sheme metapodataka. Mapiranje u današnje vrijeme predstavlja najčešću metodu kojom se omogućava interoperabilnost među različitim shemama metapodataka. Postupak mapiranja može se provesti u samom dokumentu ili se može prikazati tablično.⁶¹

Primjer jednostavnijeg aplikacijskog profila, odnosno kontejner XML Scheme za navedeni katalogni zapis (Slika 10) i primjer odgovarajućeg XML zapisa (Slika 11) prikazuju sintaksu koja se koristi prilikom kreiranja aplikacijskog profila (Slika 10). U primjeru su korištene Dublin Core (DC), VRA Core i MODS sheme metapodataka. Osim navedenih, koristi se i tzv. proširena ili kvalificirana Dublin Core shema metapodataka objedinjena pod nazivom *DCMI Metadata Terms* koja se sastoji od elemenata osnovnog i proširenog skupa Dublin Core-

⁵⁹ Usp. Isto, str. 202.

⁶⁰ Usp. Isto, str. 202.

⁶¹ Usp. Isto, str. 212-213.

a. Bez ovog standarda ne bi bilo moguće rabiti element iz proširenog skupa DC-ja *issued* (<*dcterms:issued*>) koji omogućuje istančaniju primjenu osnovnog elementa *date* iz osnovnog skupa DC-ja kako bi označio da je knjiga objavljena navedene godine. U kontejner XML Schemi *aplikacijskiprofil.xsd* navedene su metapodatkovne sheme i njima pripadni imenski prostori, putanje do tih shema, korijenski element (<*katalogni_zapis*>) te element u kojem se nalaze elementi iz navedenih shema metapodataka (<*zapis*>). U XML dokumentu *aplikacijskiprofil.xml* prikazan je sadržaj označen elementima iz navedenih shema. Na slici je vidljivo da se prefiksi pojedinih elemenata razlikuju, to jest da svaki prefiks predstavlja drugu metapodatkovnu shemu, odnosno drugi imenski prostor (Slika 11). Odabir metapodatkovnih shema bio je proizvoljan isključivo radi prikaza sintakse aplikacijskog profila.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- 1. dio: Imenski prostori (Namespaces) -->
<xs:schema targetNamespace="http://aplikacijski"
  elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="http://aplikacijski">

  <!-- 2. dio: Anotacija -->
  <xs:annotation>
    <xs:documentation xml:lang="en">
      kgagulic DC VRA Core MODS container XML Schema.
      Created 2024-08-27.
      Created by Kristina Gagulic.
    </xs:documentation>
  </xs:annotation>

  <!-- 3. dio: Uvoženje (import) drugih shema metapodataka -->
  <xs:import namespace="http://purl.org/dc/elements/1.1/" schemaLocation="https://www.dublincore.org/schemas/xmls/qdc/2003/04/02/dc.xsd" />
  <xs:import namespace="http://purl.org/dc/terms/" schemaLocation="https://www.dublincore.org/schemas/xmls/qdc/dcterms.xsd" />
  <xs:import namespace="http://www.vraweb.org/vracore4.htm" schemaLocation="https://www.loc.gov/standards/vracore/vra.xsd" />
  <xs:import namespace="http://www.loc.gov/mods/v3" schemaLocation="https://www.loc.gov/standards/mods/mods.xsd" />

  <!-- 4. dio: definiranje korijenskog elementa testne datoteke -->
  <xs:element name="katalogni_zapis">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="knjiga" maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <!-- 5. dio: definiranje elementa koji "drži" zapise -->
  <xs:element name="knjiga">
    <xs:complexType>
      <xs:sequence>
        <xs:any namespace="http://purl.org/dc/elements/1.1/" processContents="strict" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="http://purl.org/dc/terms/" processContents="strict" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="http://www.vraweb.org/vracore4.htm" processContents="strict" minOccurs="0" maxOccurs="unbounded"/>
        <xs:any namespace="http://www.loc.gov/mods/v3" processContents="strict" minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Slika 10. Prikaz kontejner XML Scheme *aplikacijskiprofil.xsd*.

```

<?xml version="1.0" encoding="UTF-8"?>
<katalogzni_zapis xmlns="http://aplikacijski"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:vra="http://www.vraweb.org/vracore4.htm"
  xmlns:mods="http://www.loc.gov/mods/v3"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://aplikacijski aplikacijskiprofil.xsd">
  <knjiga>
    <dc:identifier>101105028</dc:identifier>
    <dc:creator>Krlježa, Miroslav</dc:creator>
    <dc:description>Inv. br.: 74.712hk.</dc:description>
    <dc:subject xsi:type="dcterms:UDC">821.163.42-31</dc:subject>
    <dc:subject xsi:type="dcterms:UDC">821.163.42-8</dc:subject>
    <dc:subject>
      Psihološki roman -- Hrvatska književnost
      Socijalni roman -- Hrvatska književnost
      Egzistencijalističke pjesme -- Hrvatska književnost
    </dc:subject>
    <dc:relation>http://161.53.208.119/cgi-bin/unilib.cgi?form=D1101105028</dc:relation>
    <dcterms:issued>1932</dcterms:issued>
    <vra:title xml:lang="hrv">Povratak Filipa Latinovicza</vra:title>
    <mods:publisher>Minerva</mods:publisher>
    <mods:abstract>
      Povratak Filipa Latinovicza je roman Miroslava Krlježe objavljen 1932. godine;
      smatra se prvim cjelovitim modernim romanom u hrvatskoj književnosti.
      Po strukturi je vrlo složen, iako bez klasične kompozicije i razgranate fabule.
      <!--Tekst preuzet sa: znanje.hr-->
    </mods:abstract>
    <mods:physicalDescription>
      <mods:form>215 str.; 20 cm</mods:form>
    </mods:physicalDescription>
    <mods:location>
      <mods:physicalLocation>Knjižnica Filozofskog fakulteta Osijek</mods:physicalLocation>
    </mods:location>
  </knjiga>
</katalogzni_zapis>

```

Slika 11. Prikaz XML zapisa koji sadrži elemente više shema metapodataka
aplikacijskiprofil.xml.

5. ZAKLJUČAK

U ovom radu prikazan je značaj XML imenskih prostora kao ključnog koncepta za organizaciju sadržaja i metapodataka u XML dokumentima. XML predstavlja jedan od jezika za označavanje sadržaja, no osim sadržaja XML može označavati i metapodatke, odnosno informacije o drugim informacijama. XML također pruža mogućnost da se na temelju XML specifikacije kreiraju drugi označiteljski jezici koji se primjenjuju u određene svrhe za označavanje sadržaja povezanih uz neko područje djelovanja. Ta se mogućnost pruža zbog mogućnosti samostalnog kreiranja proizvoljnih oznaka u XML dokumentu. Te je oznake potrebno definirati u DTD-ju ili još bolje, XML Schemi. XML Schema naspram DTD-ja nudi mnogobrojne prednosti u određivanju strukture XML dokumenta.

Jedan od razloga za razvoj XML Scheme i XML imenskih prostora bio je problem podudaranje imena elemenata i atributa različitog značenja unutar XML dokumenta. XML imenski prostori, koji su podržani u XML Schemi, omogućuju da se elementi s istim prefiksom razlikuju pomoću jedinstvenog identifikatora (URI), čime se sprječava sukob naziva elemenata, bez obzira na broj različitih pravila ili shema metapodataka koje se koriste. Osim toga, XML imenski prostori osiguravaju interoperabilnost, odnosno komunikaciju između računala te 'komunikaciju' između više različitih metapodatkovnih shema, što je posebno važno u procesu mapiranja metapodataka. XML Schema i pridruženi XML imenski prostori čine organizaciju podataka znatno fleksibilnijom, opisivanje sadržaja pomoću elemenata čine detaljnijim te omogućavaju da XML zapisi budu pregledniji, što smanjuje mogućnost pogrešaka tijekom obrade podataka.

POPIS LITERATURE

1. Berners-Lee, Tim ...[et al.]. Uniform Resource Identifier (URI): Generic Syntax, 2005., str. 1. URL: <https://www.rfc-editor.org/rfc/pdf/rfc3986.txt.pdf> (2024-08-24)
2. Bosančić, Boris. XML imenski prostori i povezivanje metapodatkovnih standarda baštinskih ustanova. // 13. seminar Arhivi, knjižnice, muzeji: mogućnosti suradnje u okruženju globalne informacijske infrastrukture : zbornik radova / uredile Mirna Willer i Sanjica Faletar Tanacković. Zagreb: Hrvatsko knjižničarsko društvo, 2010 ; str. 201-220.
3. Heery, Rachel; Manjula Patel. Application Profiles : Mixing and Matching Metadata Schemas. // Ariadne 25, 9(2000). URL: <https://www.ariadne.ac.uk/issue/25/app-profiles/> (2024-09-08)
4. Kirasić, Damir. XML tehnologija i primjena u sustavima procesne informatike. // Proceedings of the 28th International Convention MIPRO 2005 / Kljaić, Josip ; Baranović, Neven (ur.). Rijeka: Hrvatska udruga za informacijsku i komunikacijsku tehnologiju, elektroniku i mikroelektroniku - MIPRO, 2005 ; str. 79-88. URL: <https://www.bib.irb.hr:8443/198358> (2024-08-24)
5. Namespaces in XML 1.0 (Third Edition). // W3C, 2009. URL: <https://www.w3.org/TR/xml-names/> (2024-08-24)
6. Pitts-Moultis, Natanya. Osnove XML : [brz i lak pristup ključnim vladajućim veštinama i tehnologijama]. Čačak : Kompjuter biblioteka, 2000.
7. Roy, Jaideep; Ramanujan, Anupama. XML schema language: taking XML to the next level. // IT Professional 2, 3(2001) ; str. 37-40. URL: <https://doi.org/10.1109/6294.918217> (2024-08-24)
8. What is a Namespace. URL: <https://www.programmingcube.com/what-is-a-namespace/> (2024-08-24)
9. XML Namespaces. // W3Schools. URL: https://www.w3schools.com/xml/xml_namespaces.asp (2024-08-24)
10. XML Schema // W3C, 2000. URL: <https://www.w3.org/XML/Schema> (2024-09-08)
11. XML Schema. // W3Schools. URL: https://www.w3schools.com/xml/xml_schema.asp (2024-09-08).

12. XML. // Encyclopaedia Britannica Online. Encyclopedia Britannica, 2024. URL:
<https://www.britannica.com/technology/XML> (2024-08-24)