

Infrastruktura za elektronsko poslovanje (SI3IEP)

Nemanja Kojić

XML

eXtensible Markup Language

Uvod

- XML je skraćenica od tri reči:
e**X**tensible **M**arkup **L**anguage.
- XML je dizajnirana za potrebe:
opisa,
prenosa i
čuvanja podataka
- Veoma je važno znati XML
- Lako se uči

```
<?xml version="1.0"?>
<note>
  <to>John</to>
  <from>Marry</from>
  <heading>Reminder</heading>
  <body>Let's talk tomorrow.</body>
</note>
```

Uvod

- **Šta je XML?**
 - XML označava EXtensible Markup Language
 - XML je jezik za označavanje (veoma sličan jeziku HTML)
 - XML je dizajniran za potrebe prenosa podataka (ne za njihovo prikazivanje)
 - XML tagovi nisu predefinisani (korisnik treba da definiše sopstvene)
 - XML je dizajniran tako da bude samoopisiv
- XML omogućava prenos podataka koji je nezavisan od hardverske i softverske platforme

Uvod

(razlika između XML-a i HTML-a)

- XML nije zamena za HTML.
- XML i HTML su stvoreni na osnovu različitih ciljeva:
 - XML je dizajniran za prenos i čuvanje podataka (akcentat je na vrsti i značenju podataka)
 - HTML je dizajniran za prikazivanje podataka (akcentat je na tome kako podaci izgledaju)
- HTML se odnosi na prikazivanje informacija, a XML se odnosi na prenos informacija.

Uvod

- XML omogućava:
 - Strukturiranje podataka
 - Prenos podataka
 - Čuvanje podataka
- Dati XML dokument predstavlja čiste informacije koje su uokvirene u tagove
- Dokument sam sebe opisuje
- Poseban softver se može napisati za potrebe slanja, prijema i prikazivanja poruka.
- Format poruka je zapisan u XML dokumentu
- Imena tagova impliciraju značenje podataka

```
<?xml version="1.0"?>
<note>
  <to>John</to>
  <from>Marry</from>
  <heading>Reminder</heading>
  <body>Let's talk tomorrow.</body>
</note>
```

Uvod

- XML dokument je tekstualni dokument (kao i svaki drugi tekstualni dokument)
- Aplikacije koje procesiraju XML dokumente tagove tretiraju na specijalan način
- Značenje tagova zavisi od prirode aplikacije i domena problema
- Tagovi nisu propisani XML standardom
- Korisnik kreira sopstvene tagove (koji odgovaraju elementima domena problema)
- XML nema predefinisane tagove
- XML nema ni predefinisanu strukturu (korisnik može da dizajnira proizvoljnu strukturu XML dokumenta prema sopstvenim potrebama)

Primeri korišćenja

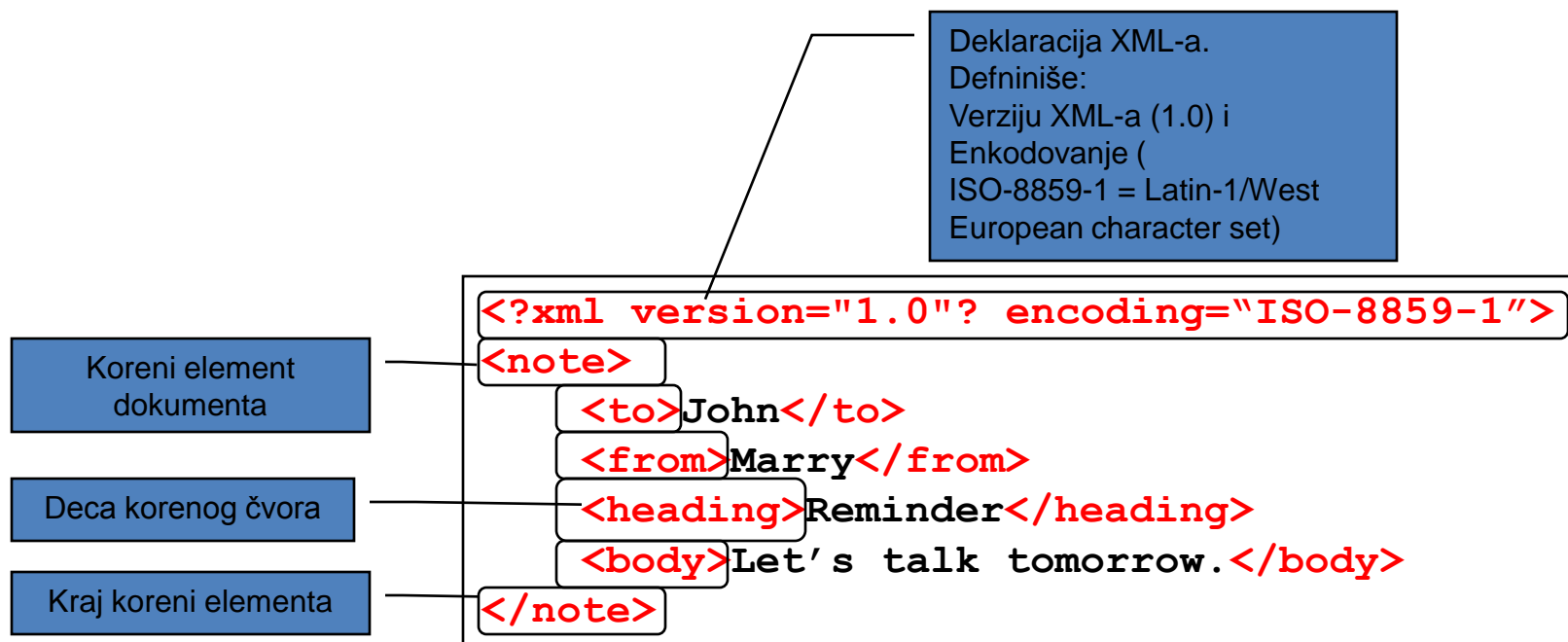
- XML razdvaja podatke od HTML-a
 - Prikazivanje dinamičkih podataka bez promena HTML-a
 - Promenljive informacije su u XML fajlovima
 - JavaScript omogućava jednostavno čitanje ovih podataka i njihovo prikazivanje
- XML pojednostavljuje deljenje podataka
 - Rešava problem nekompatibilnih formata podataka koje koriste različiti sistemi i baze podataka
 - Informacije se čuvaju u formatu koji ne zavisi od platforme ili baza podataka
- XML pojednostavljuje razmenu podataka
 - Rešava problem razmene podataka između različitih nekompatibilnih sistema na jednostavan način

Primeri korišćenja

- XML pojednostavljuje promene platforme
 - Migracija podataka, konverzija u drugi format (može dovesti do gubitka nekomatibilnih podataka)
 - XML se podaci čuvaju u tekstualnom formatu (lako ih je prilagoditi potrebama nove platforme ili nove verzije platforme)
- XML se koristi na pravljenje novih Internet jezika:
 - Primeri nekih jezika:
 - XHTML poslednja verzija HTML-a
 - WSDL za opis veb servisa
 - WAP i WML – za pravljenje sadržaja koji se mogu pregledati mobilnim uređajima
 - RSS za dobijanje vesti
 - RDF i OWL – za opis resursa i ontologije (konceptata iz stvarnog sveta)
 - SMIL za opis multimedije na webu

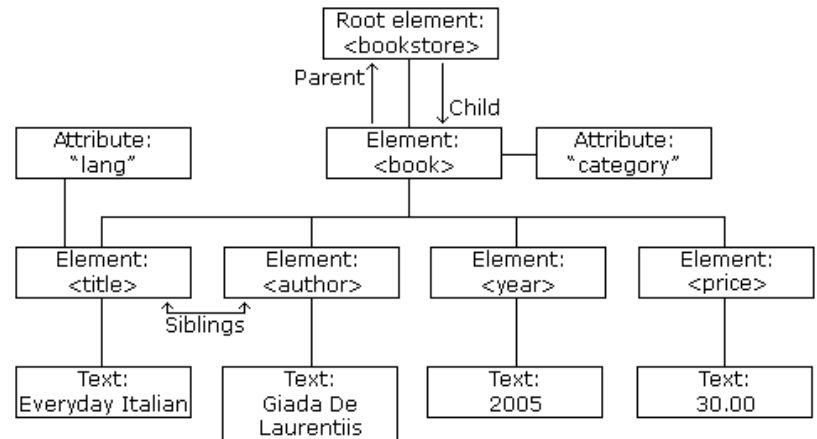
Struktura XML dokumenta

- XML dokument ima strukturu stabla
 - Jedan koreni čvor i njegovi podčvorovi
- XML dokument mora da ima koreni element



XML primer (1)

```
<bookstore>
  <book category="COOKING">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category="CHILDREN">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category="WEB">
    <title lang="en">Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>
```



XML sintaksna pravila

- Svi elementi XML-a moraju imati tagove zatvaranja
- XML tagovi su “case sensitive”
- XML elementi moraju biti pravilno ugneždeni
- XML dokumenti moraju imati koreni element
- Vrednosti atributa XML elemenata moraju biti napisane pod znacima navoda (“...”)
- Komentari: `<!-- Ovo je komentar -->`
- Beli znakovi su deo sadržaja XML dokumenta (ne zanemaruju se)

Reference na entitete (*Entity References*)

- Neki XML znakovi imaju specijalno značenje (specijalni karakteri)
- To su: < > ' " &
- Kako predstaviti sledeći sadržaj?
 - `<message>if salary < 1000 then</message>`
- Karakter < se zamenjuje referencom na odgovarajući entitet:
 - `<message>if salary < 1000 then</message>`
- XML ima 5 predefinisanih entiteta:

<code>&lt;</code>	<
<code>&gt;</code>	>
<code>&amp;</code>	&
<code>&apos;</code>	'
<code>&quot;</code>	"

Elementi i atributi

- Element čini sadržaj između početnog i zatvarajućeg taga (uključujući i njih)
- Element može sadržati:
 - Samo Običan tekst
 - Samo Druge elemente (deca)
 - I tekst i elemente
- Preporuke imenovanja elemenata (tagova)
 - Koristiti deskriptivne nazive (<first_name>)
 - Nazivi treba da budu jednostavni (i kratki)
 - Izbegavati sledeće karaktere za formiranje naziva tagova
dvotačku (:),
tačku (.)
minus/crtica (-)

Atributi elemenata

- Elementi mogu imati attribute
- Atributi se specificiraju u okviru startnog taga odgovarajućeg elementa
- Atributi najčešće nose dodatne informacije o samom elementu
koje nisu konkretni podaci
- Primer: tip fajla je irelevantan za same podatke, ali je važan za softver koji procesira element

```
<file type="gif">computer.gif</file>
```

Preporuke za korišćenje atributa elementi ili atributi?

- I atributi i (pod)elementi nose istovetne informacije o elementima
- Izbegavati preterano korišćenje atributa
 - Atributi ne mogu sadržati više od jedne vrednosti (elementi mogu)
 - Atributi ne mogu da sadrže informacije u obliku stabla (elementi mogu)
 - Atributi nisu lako proširivi (za buduće promene)
- Attribute koristiti za metapodatke (npr. Uvesti identifikator XML elemenata za njihovo jedinstveno identifikovanje u dokumentu)
- Metapodaci su podaci o podacima

```
<?xml version="1.0"?>
<note id="100">
  <to>John</to>
  <from>Marry</from>
  <heading>Reminder</heading>
  <body>Let's talk tomorrow.</body>
</note>
```

Validacija XML dokumenta

- XML fajlovi moraju biti pravilno formirani (*well-formed*)
- Dobro formirani XML fajlovi moraju biti sintaksno ispravni
- Mogu se definisati pravila formata (strukture) XML dokumenta
- Ako postoje ova pravila, sadržaj dobro formiranog XML dokumenta treba da bude u skladu sa tim pravilima

Opis strukture XML dokumenta

XML DTD (*Document Type Def.*)

- Definiše strukturu XML dokumenta
- Definiše i skup legalnih elemenata
- Primer:

```
<!DOCTYPE note
[
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
]>
```

XML Schema (alternativa DTD-u)

```
<xs:element name="note">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="to" type="xs:string"/>
      <xs:element name="from" type="xs:string"/>
      <xs:element name="heading" type="xs:string"/>
      <xs:element name="body" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

XML

Napredni pojmovi

Prostori imena (*Namespaces*)

- Metod izbegavanja kolizije imena
- Do kolizije dolazi kada se mešaju XML dokumenti iz različitih aplikacija (koriste iste elemente, ali sa različitim značenjem i strukturom)
- Kolizije se rešavaju prefiksima

```
<h:table>
  <h:tr>
    <h:td>Apples</h:td>
    <h:td>Bananas</h:td>
  </h:tr>
</h:table>

<f:table>
  <f:name>African Coffee Table</f:name>
  <f:width>80</f:width>
  <f:length>120</f:length>
</f:table>
```

XML Namespaces (xmlns)

- Da bi se koristili prefiksi, za njih treba da postoje definicije tzv. prostora imena
- Prostor imena se definiše vrednošću taga xmlns u okviru startnog taga proizvoljnog elementa
- Sintaksa za deklaraciju prostora imena:

`xmlns:prefix="URI"`

```
<root>
<h:table xmlns:h="http://www.w3.org/TR/html4/">
  <h:tr>
    <h:td>Apples</h:td>
    <h:td>Bananas</h:td>
  </h:tr>
</h:table>

<f:table xmlns:f="http://www.w3schools.com/furniture">
  <f:name>African Coffee Table</f:name>
  <f:width>80</f:width>
  <f:length>120</f:length>
</f:table>
</root>
```

Enkodovanje (encoding)

- XML dokumenti mogu sadržati i karaktere koji ne pripadaju skupu ASCII znakova
- Da bi se izbegle greške pri čitanju, specificira se enkodovanje dokumenta
- Dve vrsta grešaka:
 - Nevalidni karakter u XML dokumentu (neASCII znak u dokumentu koji je sačuvan kao jednobajtni ASCII bez atributa *encoding*)
 - Prelazak sa jednog tipa enkodovanja na drugi nije podržan (npr. XML fajl je sačuvan sa enkodovanje UTF-16, a u XML dokumentu je specificiran 1-bajtno enkodovanje UTF-8)
- Preporuka je da se uvek specificira enkodovanje i da se koriste editori XML-a koji podržavaju različita enkodovanja

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>John</to>
  ...
</note>
```

XML DOM

- XML DOM (Document Object Model)
- Definiše standard za pristup i manipulisanje XML dokumentima
- Vidi XML dokumente kao stablo elemenata
- Elementi, atributi i tekst se označavaju kao čvorovi
- Najčešće korišćene XML DOM metode za manipulisanje XML dokumentima:
 - Dohvatanje vrednosti elementa
 - Dohvatanje vrednosti atributa
 - Promena vrednosti elementa
 - Promena vrednosti atributa
 - Pravljenje novog elementa
 - Uklanjanje datog elementa

XML tehnologije

- [XHTML \(Extensible HTML\)](#)
A stricter and cleaner XML based version of HTML.
- [XML DOM \(XML Document Object Model\)](#)
A standard document model for accessing and manipulating XML.
- [XSL \(Extensible Style Sheet Language\)](#) XSL consists of three parts:
- [XSLT \(XSL Transform\)](#) - transforms XML into other formats, like HTML
- [XSL-FO \(XSL Formatting Objects\)](#)- for formatting XML to screen, paper, etc
- [XPath](#) - a language for navigating XML documents
- [XQuery \(XML Query Language\)](#)
An XML based language for querying XML data.
- [DTD \(Document Type Definition\)](#)
A standard for defining the legal elements in an XML document.
- [XSD \(XML Schema\)](#)
An XML-based alternative to DTD.
- [XLink \(XML Linking Language\)](#)
A language for creating hyperlinks in XML documents.
- [XPointer \(XML Pointer Language\)](#)
Allows the XLink hyperlinks to point to more specific parts in the XML document.
- [XForms \(XML Forms\)](#)
Uses XML to define form data.
- [SOAP \(Simple Object Access Protocol\)](#)
An XML-based protocol to let applications exchange information over HTTP.
- [WSDL \(Web Services Description Language\)](#)
An XML-based language for describing web services.
- [RDF \(Resource Description Framework\)](#)
An XML-based language for describing web resources.
- [RSS \(Really Simple Syndication\)](#)
A format for syndicating news and the content of news-like sites.
- [WAP \(Wireless Application Protocol\)](#)
A XML based language for displaying content on wireless clients, like mobile phones.
- [SMIL \(Synchronized Multimedia Integration Language\)](#)
A language for describing audiovisual presentations.
- [SVG \(Scalable Vector Graphics\)](#)
Defines graphics in XML format.

XML – zaključak

- XML se koristi za deljenje, razmenu i čuvanje podataka
- XML je nezavisan od platforme
- XML dokumenti imaju strukturu stabla
 - Postoji obavezno koreni čvor
- XML se odlikuje veoma jednostavnim sintaksnim pravilima.
- XML sa korektno sintaksom je “dobro formiran”.
- Validacija strukture XML fajla može da se sprovede i na osnovu DTD šeme
- [XSLT](#) se koristi za transformaciju XML-a u druge formate (HTML).
- Svi moderni pretraživači imaju ugrađene parsere za XML koji omogućavaju čitanje i manipulisanje XML dokumentima
- XML Namespace je način izbegavanja kolizije imena u XML dokumentima
- Parser XML-a ignoriše tekst unutar [CDATA](#) sekcije.

Parsiranje XML-a

- Postoje dve vrste parsera za XML
 - DOM parseri
 - SAX parseri
- DOM parseri učitavaju celu strukturu XML u memoriju
 - pogodni su za manje XML fajlove
 - omogućavaju pristup svim elementima u bilo kom trenutku
 - laki za upotrebu
- SAX parseri ne rade nad celim XML fajlom
 - parsiraju “liniju po liniju”
 - implementiranje obrade pojedinačnih događaja prepoznavanja elemenata
 - pogodniji su za veće XML fajlove
 - ne zauzimaju puno memorije
 - brzi su
 - malo su teži za upotrebu