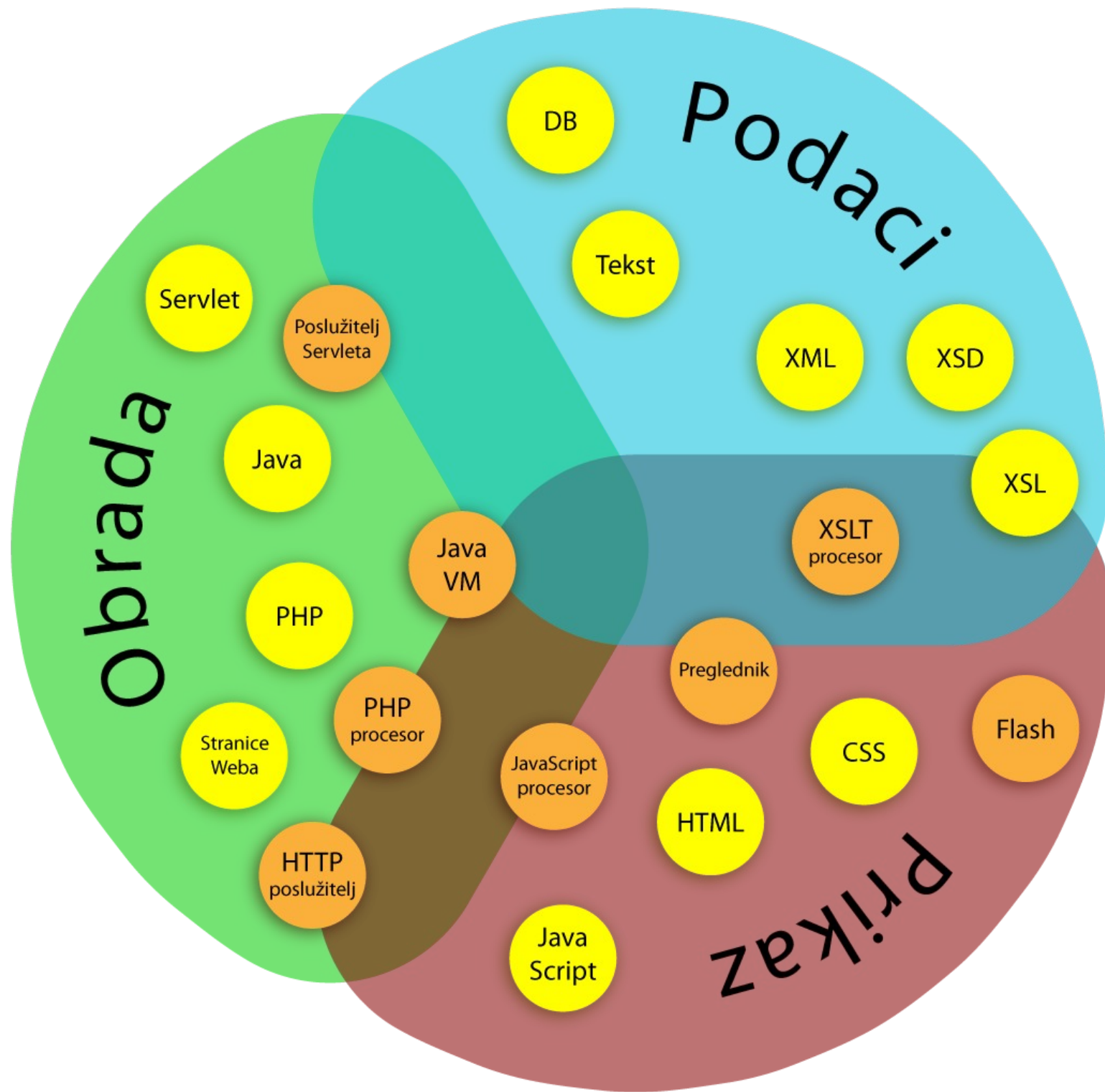
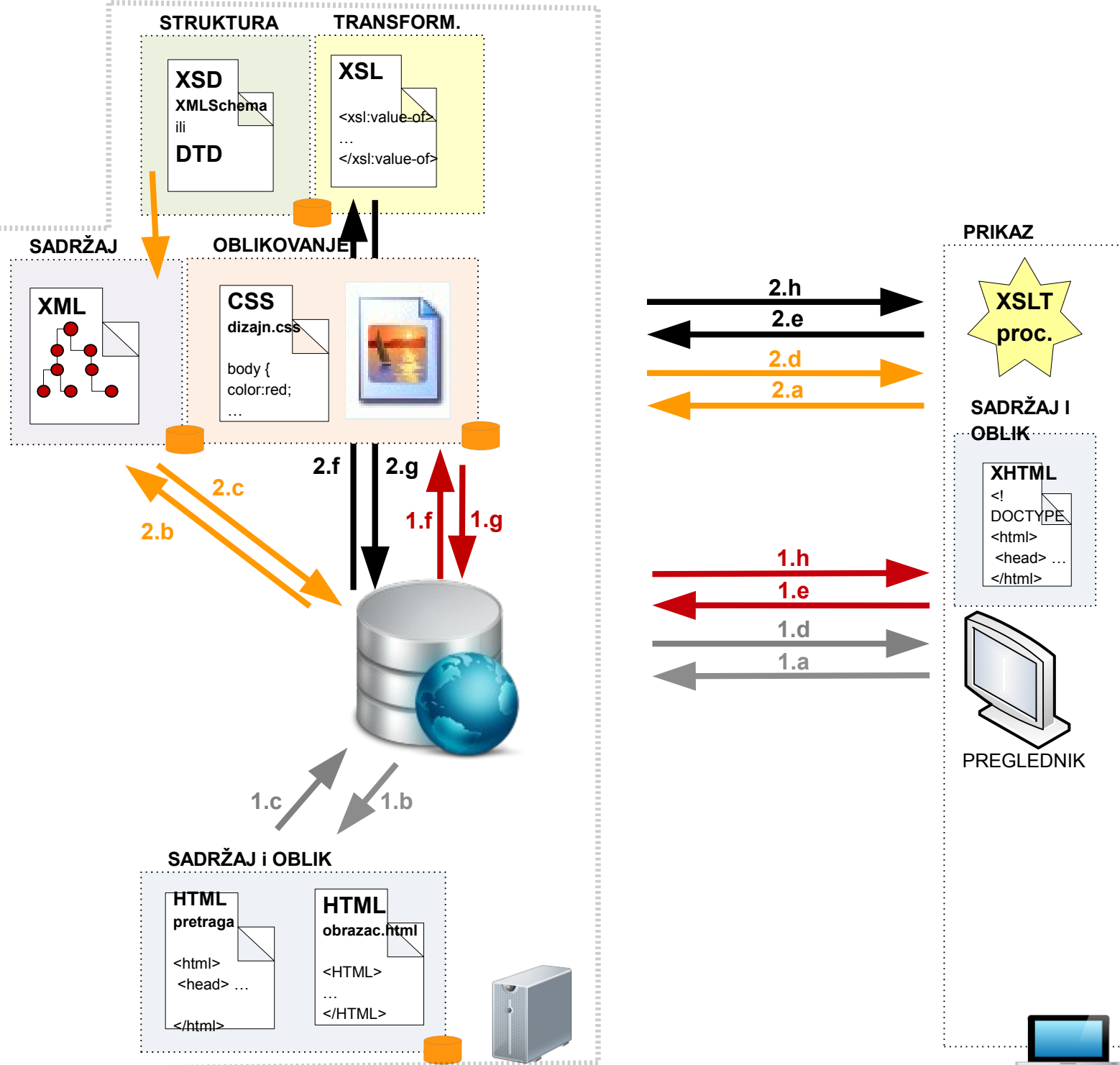


2. laboratorijska vježba

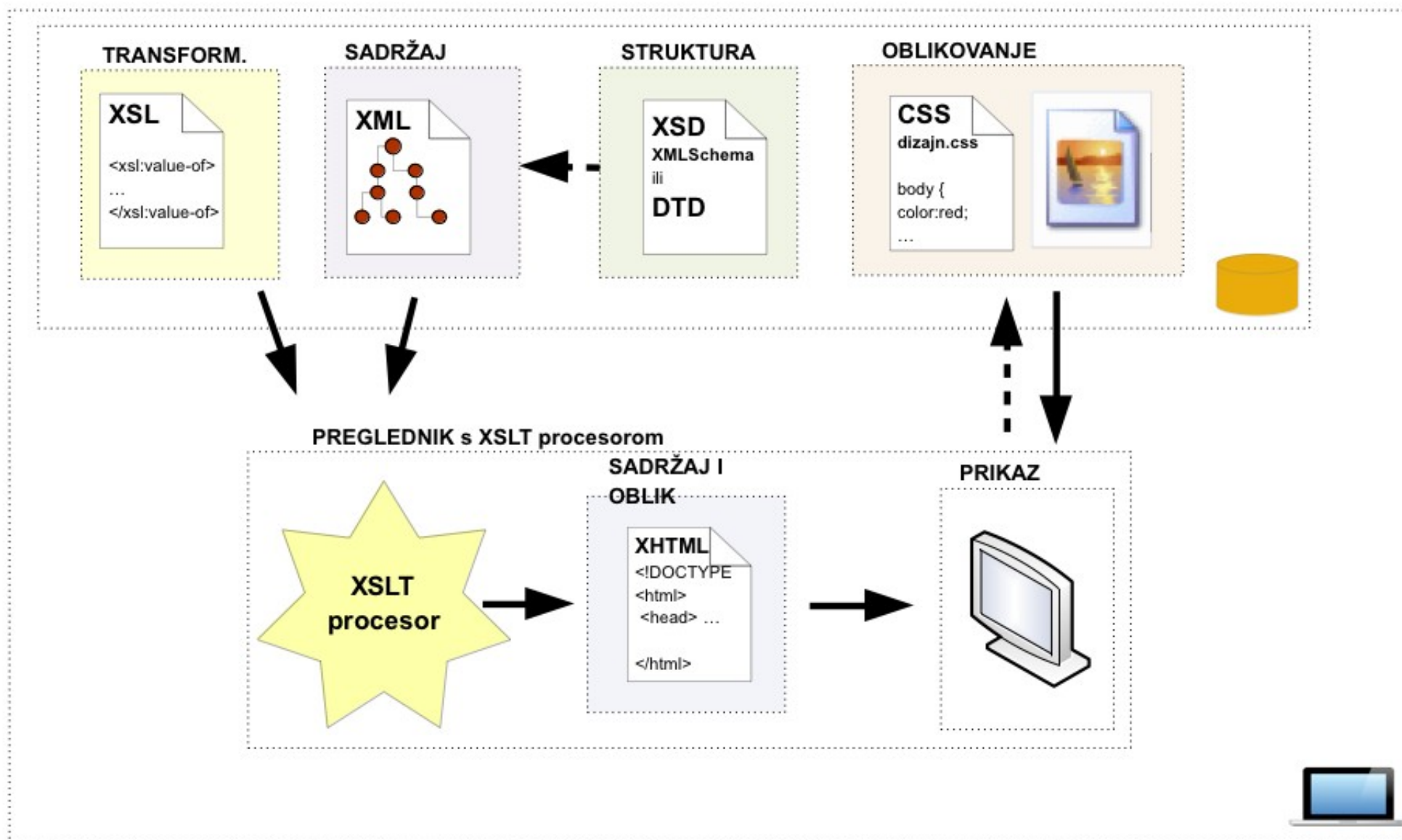
- XML
- DTD
- XSL



2. vježba



2. vježba



Domaća zadaća

- Proučiti osnove jezika XML i pisanja dokumenata
<http://www.w3schools.com/xml/>
- Proučiti osnove pisanja DTD-a
<http://www.w3schools.com/dtd/>
- Proučiti osnove XSL-a, način povezivanja XML dokumenta s XSL stilovima i korištenje kontrolnih struktura XSL-a
<http://www.w3schools.com/xsl/>
<http://nwalsh.com/docs/tutorials/xsl/>

Ciljevi vježbe

- Upoznavanje sa strukturiranjem podataka kroz XML
- Razlikovanje pojmova dobro oblikovanog i valjanog XML dokumenta
- Opis strukture XML dokumenta korištenjem DTD-a
- Priprema prikaza podataka iz XML dokumenta korištenjem jezika XSL
- Za sve ovo koristit ćemo rezultate prve vježbe
 - Ne krećemo *od nule* :)

Zadatak vježbe



- Izraditi:
 - Vlastitu strukturu podataka
 - XML dokument s **barem** 5 podataka prve razine
 - (5 knjiga/5 osoba u imeniku, itd.)
 - DTD dokument s opisom strukture
 - za provjeru valjanosti XML dokumenta
 - XSL dokument za prikaz podataka iz XML dokumenta u obliku (X)HTML stranice
 - Prema vašem rješenju prve vježbe
 - CSS – već ga imamo



Tablica strukture podataka

- redaka treba biti barem 15
- mora postojati jedinstveni identifikator pojedinog zapisa
- mora postojati Facebook ID/name – identifikator zapisa na Facebooku
 - (trebate ga sami ručno pronaći)
- Barem 4 hijerarhijske razine strukturiranih podataka
- Barem 2 elementa brojnosti N
- Barem 2 atributa
- Barem 2 elementa ne moraju nužno postojati u određenom zapisu

Tablica strukture podataka



REDNI BROJ	NAZIV	HIJERARHIJSKA RAZINA	BROJNOST	ELEMENT I ATRIBUT	OBAVEZNOST POSTOJANJA	VRIJEDNOSTI	PRIMJER: SLOBODAN UPIS VRIJEDNOSTI ili SKUP DOZVOLJENIH VRIJEDNOSTI	ELEMENT U OBRASCU na stranici za pretraživanje
1	Osoba	1	N	E	DA	NEMA		
2	OIB	2	1	A	DA	DA	11 znamenaka	POLJE ZA UNOS
3	Ime	2	1	E	DA	DA	npr. Ivo	POLJE ZA UNOS
4	Prezime	2	1	E	DA	DA	npr. Ivić	POLJE ZA UNOS
5	Kategorija	2	1	A	DA	SKUP	Prijatelj, Kolega, Obitelj, Bez kategorije	KVADRATIĆ ZA IZBOR (checkbox) 4 kategorije
6	Telefon	2	N	E	DA	NEMA		
7	Tip	3	1	A	DA	SKUP	Mobilni, Fiksni, Telefaks	KRUŽIĆ ZA ODABIR (radio) jednog tipa
8	Broj	3	1	E	DA	DA	npr. 6129999	POLJE ZA UNOS
9	Pozivni broj mreže	4	1	A	DA	SKUP	099, 098, 095, 092, 091, 01	IZBORNİK ZA VIŠESTRUKI ODABIR (multiple select) poziv. broja
10	Adresa	2	1	E	NE	NEMA		KVADRATIĆ ZA IZBOR (checkbox) postojanja adrese
11	Ulica	3	1	E	DA	DA	npr. Ilica	POLJE ZA UNOS
12	Kućni broj	3	1	E	DA	DA	npr. 111	POLJE ZA UNOS
13	Mjesto	3	1	E	DA	DA	npr. Zagreb	POLJE ZA UNOS
14	Poštanski broj	4	1	A	DA	DA	npr. 10000	POLJE ZA UNOS
15	Država	3	1	E	DA	DA	npr. Hrvatska	POLJE ZA UNOS
16	Mail adresa	2	N	E	NE	DA	npr. ivo.ivic@fer.hr	POLJE ZA UNOS KVADRATIĆ ZA IZBOR (checkbox) postojanja mail adrese
17	PROIZVOLJNI ODABIR							
18	PROIZVOLJNI ODABIR							
19	PROIZVOLJNI ODABIR							
20	PROIZVOLJNI ODABIR							

Struktura podataka - primjer



- Podaci iz meteoroloških postaja
 - Naziv postaje i e-mail adresa
 - Smjer i brzina vjetra
 - Temperatura, vlažnost i tlak zraka
 - Vrijeme (naoblaka, padaline, ...)



Korak 1: XML dokument

- Struktura stabla
- Sastoji se od elemenata
 - Imaju pridružene attribute
 - Sadrže druge elemente, proizvoljan sadržaj ili su prazni
- Bitno: razlučiti *mjesto* podatka
 - U element?
 - U atribut?
 - U sadržaj?
- Osnovno pravilo (*rule of thumb*)
 - Podaci – u elemente
 - Prazni elementi se mogu koristiti kao zastavice (ima/nema)
 - Meta-podaci – u attribute

Primjer XML-a

- Počinje s `<?xml ...?>`
- Korijski element
 - `<podaci>`
- Lista elemenata
 - `<postaja>`
- Podelementi
 - `<naziv>`
 - `<vjetar>`
 - `<temperatura>`
 -
- Atributi
 - `<vjetar smjer="...">`

```

1  <?xml version="1.0" encoding="UTF-8" ?>
2  <podaci>
3      <postaja>
4          <naziv>Zagreb</naziv>
5          <vjetar smjer="NE">
6              <brzina>4</brzina>
7          </vjetar>
8          <temperatura>20</temperatura>
9          <vlaznost>80</vlaznost>
10         <tlak>1027</tlak>
11         <vrijeme>potpuno oblačno</vrijeme>
12     </postaja>
13     <postaja>
14         <naziv>Dubrovnik</naziv>
15         <vjetar smjer="SE">
16             <brzina>10</brzina>
17         </vjetar>
18         <temperatura>15</temperatura>
19         <vlaznost>70</vlaznost>
20         <tlak>1011</tlak>
21         <vrijeme>sunčano</vrijeme>
22     </postaja>
23 </podaci>
  
```

Najčešće pogriješke



- Nepravilno ugniježđivanje
 - primjer: `<i>tekst</i></i>` umjesto `<i>tekst</i>`
- Nezatvoreni element
 - primjer: `tekst` umjesto `tekst`
 - primjer: `
` umjesto `
`
- Oznake i atributi se pišu i velikim slovima
 - primjer: `tekst` umjesto `tekst`
 - primjer: `tekst`
- Dokument ne sadrži samo jedan korijenski element
 - primjer: Nepostojanje samo jednog `<html>` elementa



Najčešće pogreške #2



- Bez navodnika za vrijednosti atributa
 - primjer: `<table width=100%>` umjesto `<table width="100%">`
- Uporaba specijalnih znakova
 - primjer: `Ti & ja` umjesto `Ti & ja`
 - Zabranjeno je korištenje `< i &`



Provjera oblikovanosti XML-a

- Dobro oblikovani (*well-formed*) XML dokument je samo sintaksno ispravan
- Provjerite na:
 - http://www.w3schools.com/xml/xml_validator.asp
 - XMLValidator – na stranici predmeta
 - <http://www.xmlvalidation.com/>
 - ... neki drugi :)

Korak 2: DTD

- Definicija strukture dokumenta
- Sadrži niz deklaracija u proizvoljnom redoslijedu

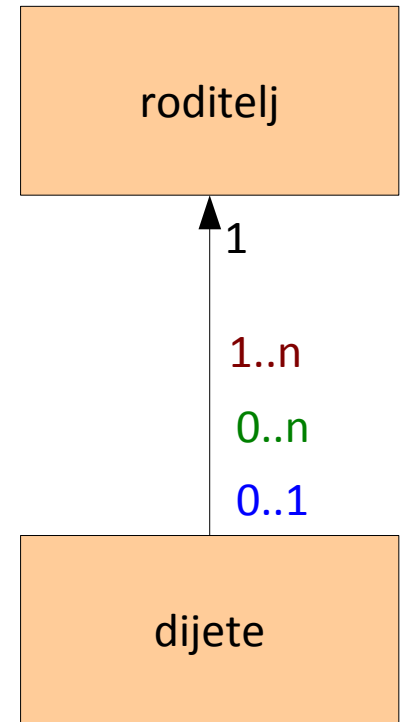
<!ELEMENT ime-elementa sadržaj>

<!ATTLIST ime-elementa ime-atributa uvjeti>

Indikatori kardinalnosti elementa



- Točno jedan element – ne navodi se indikator (prethodni primjer)
- Najmanje jedan element – indikator **+**
<!ELEMENT roditelj (dijete+)>
- Nijedan ili više elemenata – indikator *****
<!ELEMENT roditelj (dijete*)>
- Nijedan ili jedan element – indikator **?**
<!ELEMENT roditelj (dijete?)>



Kombiniranje više elemenata djece



- Operator slijeda elemenata ','
 - u dokumentu moraju biti navedeni u definiranom slijedu:
`<!ELEMENT roditelj (dijete1, dijete2, dijete3)>`
- Operator alternative (XOR) '|'
 - ili jedan ili drugi element
`<!ELEMENT roditelj (dijete1 | dijete3)>`
- Operator grupiranja ()
 - niz elemenata unutar zagrada tretira se kao jedan element kod primjene ostalih operacija i indikatora
`<!ELEMENT roditelj ((dijete1 | dijete2)+, dijete3)>`



CDATA tip atributa

- Neparsirani niz znakova

```
<!ELEMENT course (title+, lecturer, contents?)>
<!ATTLIST course status CDATA "undergrad">
```

- Ako je vrijednost atributa status navedena:

```
<course status="module"> ... </course>
```

- Ako vrijednost atributa nije navedena, nakon parsiranja atributu će biti pridjeljena vrijednost 'undergrad'

```
<course> ... </course>
```

nakon parsiranja:

```
<course status="undergrad"> ... </course>
```



Enumerirani tip atributa

- Atribut može poprimiti samo jednu od vrijednosti navedenih u deklaraciji enumeriranog tipa

`<!ELEMENT course (title+, lecturer, contents?)>`

`<!ATTLIST course status (undergrad|orientation|graduate|postgrad|module|elective|external) "undergrad">`

Ako je vrijednost atributa *status* navedena:

`<course status="module"> ... </course>`

Ako vrijednost atributa nije navedena, nakon parsiranja atributu će biti pridjeljena vrijednost 'undergrad'

`<course> ... </course>`



Podrazumijevana vrijednost



- Ako je vrijednost atributa navedena u XML dokumentu, nakon parsiranja atribut poprima tu, navedenu vrijednost
- Ako vrijednost atributa nije navedena, nakon parsiranja atributu će biti pridjeljena podrazumijevana vrijednost

```
<!ELEMENT course (title+, lecturer, contents?)>  
<!ATTLIST course status CDATA 'opći'>
```

```
<course> ... </course>
```

```
<course status="module"> ... </course>
```



Obavezna vrijednost



- Vrijednost atributa se mora navesti u XML dokumentu
- Izostavljanje vrijednosti rezultira grješkom prilikom parsiranja dokumenta

```
<!ELEMENT course (title+, lecturer, contents?)>  
<!ATTLIST course status CDATA #REQUIRED>
```

```
<course> ... </course>      greška/grješka :-) !
```

```
<course status="undergrad"> ... </course>
```



Neobavezna vrijednost

- Vrijednost atributa se može navesti u XML dokumentu
- Izostavljanje vrijednosti rezultira nepostojanjem tog atributa nakon parsiranja

```
<!ELEMENT course (title+, lecturer, contents?)>
<!ATTLIST course status CDATA #IMPLIED>
```

```
<course> ... </course>
<course status="module"> ... </course>
```



Nepromjenljiva vrijednost

- Vrijednost atributa se može navesti u dokumentu
- Ako se navede, mora se koristiti podrazumijevana vrijednost navedena u deklaraciji atributa
 - navođenje različite vrijednosti - grješka u parsiranju
 - nakon parsiranja atribut uvijek ima podrazumijevanu vrijednost

```
<!ELEMENT course (title+, lecturer, contents?)>
<!ATTLIST course status CDATA #FIXED "undergrad">
```

```
<course> ... </course>
```

```
<course status="undergrad"> ... </course>
```

```
<course status="module"> ... </course> grješka!
```



Primjer DTD-a

```

1  <!ELEMENT podaci (postaja)*>
2  <!ELEMENT postaja (naziv, vjetar, temperatura, vlaznost, tlak, vrijeme)>
3  <!ELEMENT naziv (#PCDATA)>
4  <!ELEMENT vjetar (brzina)>
5  <!ATTLIST vjetar smjer (N|NE|E|SE|S|SW|W|NW) #REQUIRED>
6  <!ELEMENT brzina (#PCDATA)>
7  <!ELEMENT temperatura (#PCDATA)>
8  <!ELEMENT vlaznost (#PCDATA)>
9  <!ELEMENT tlak (#PCDATA)>
10 <!ELEMENT vrijeme (#PCDATA)>

```

Primjer DTD-a

```

1  <!ELEMENT podaci (postaja)*>
2  <!ELEMENT postaja (naziv, vjetar, temperatura, vlaznost, tlak, vrijeme)>
3  <!ELEMENT naziv (#PCDATA)>
4  <!ELEMENT vjetar (brzina)>
5  <!ATTLIST vjetar smjer (N|NE|E|SE|S|SW|W|NW) #REQUIRED>
6  <!ELEMENT brzina (#PCDATA)>
7  <!ELEMENT temperatura (#PCDATA)>
8  <!ELEMENT vlaznost (#PCDATA)>
9  <!ELEMENT tlak (#PCDATA)>
10 <!ELEMENT vrijeme (#PCDATA)>

```

- *podaci* mogu sadržavati proizvoljan broj *postaja*
- *postaja* sadrži *naziv*, *vjetar*, ... tim redoslijedom
- *vjetar* ima obavezan atribut *smjer* iz skupa
- Ostali elementi sadrže parsirane podatke

Povezivanje XML-a sa DTD-om

- DOCTYPE direktiva u XML dokumentu
- Unutarnji DTD

`<!DOCTYPE korijen [...]>`

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

`<!DOCTYPE podaci [`

`<!ELEMENT podaci (postaja)*>`

`<!ELEMENT postaja (naziv, vjetar, temperatura, vlaznost, tlak, vrijeme)> ...`

`]>`

- Vanjski DTD

`<!DOCTYPE korijen SYSTEM url>`

`<?xml version="1.0" encoding="UTF-8" standalone="no"?>`

`<!DOCTYPE podaci SYSTEM "vrijeme.dtd">`

Provjera valjanosti XML-a

- Valjani (*valid*) XML dokument je i sintaksno ispravan i pisan prema pravilima DTD-a
- Alat za validaciju XML-a
 - XMLValidator – na stranici predmeta
 - <http://www.xmlvalidation.com/>

Korak 3: XSL



- Cilj: XSL transformacija XML dokumenta koja daje HTML stranicu s tabličnim prikazom podataka

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <podaci>
3   <postaja>
4     <naziv>Zagreb</naziv>
5     <vjetar smjer="NE">
6       <brzina>4</brzina>
7     </vjetar>
8     <temperatura>20</temperatura>
9     <vlaznost>80</vlaznost>
10    <tlak>1027</tlak>
11    <vrijeme>potpuno oblačno</vrijeme>
12  </postaja>
13  <postaja>
14    <naziv>Dubrovnik</naziv>
15    <vjetar smjer="SE">
16      <brzina>10</brzina>
17    </vjetar>
18    <temperatura>15</temperatura>
19    <vlaznost>70</vlaznost>
20    <tlak>1011</tlak>
21    <vrijeme>sunčano</vrijeme>
22  </postaja>
23 </podaci>
```





VREMENSKA PROGNOZA

Početna Pretraživanje FER e-Campus Kontakt

Naziv	Smjer vjetra	Brzina vjetra	Temperatura	Vlažnost	Tlak	Vrijeme	Mail adresa postaje
Zagreb	NE	4	20	80	1027	potpuno oblačno	Zagreb
Dubrovnik	SE	10	15	70	1011	sunčano	Dubrovnik

Footer

XSL dokument

- XSL transformira *stablo* originalnog XML dokumenta u novo stablo koje ispisuje kao izlaz
- XSL je XML dokument s korijenskim elementom *stylesheet* iz prostora imena *xsl/* (u ovom slučaju):

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

```
<xsl:stylesheet
```

```
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
```

```
version="1.0">
```

```
...
```

```
</xsl:stylesheet>
```

Podešavanje izlaza

- DOCTYPE pridružen izlazu postavlja se sa

```
<xsl:output method="xml" indent="yes"
  doctype-system="http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"
  doctype-public="-//W3C//DTD XHTML 1.0 Strict//EN" />
```

ovo odgovara deklaraciji u izlazu

```
<!DOCTYPE prvi_element PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

XHTML!

Povezivanje XML-a sa XSL-om

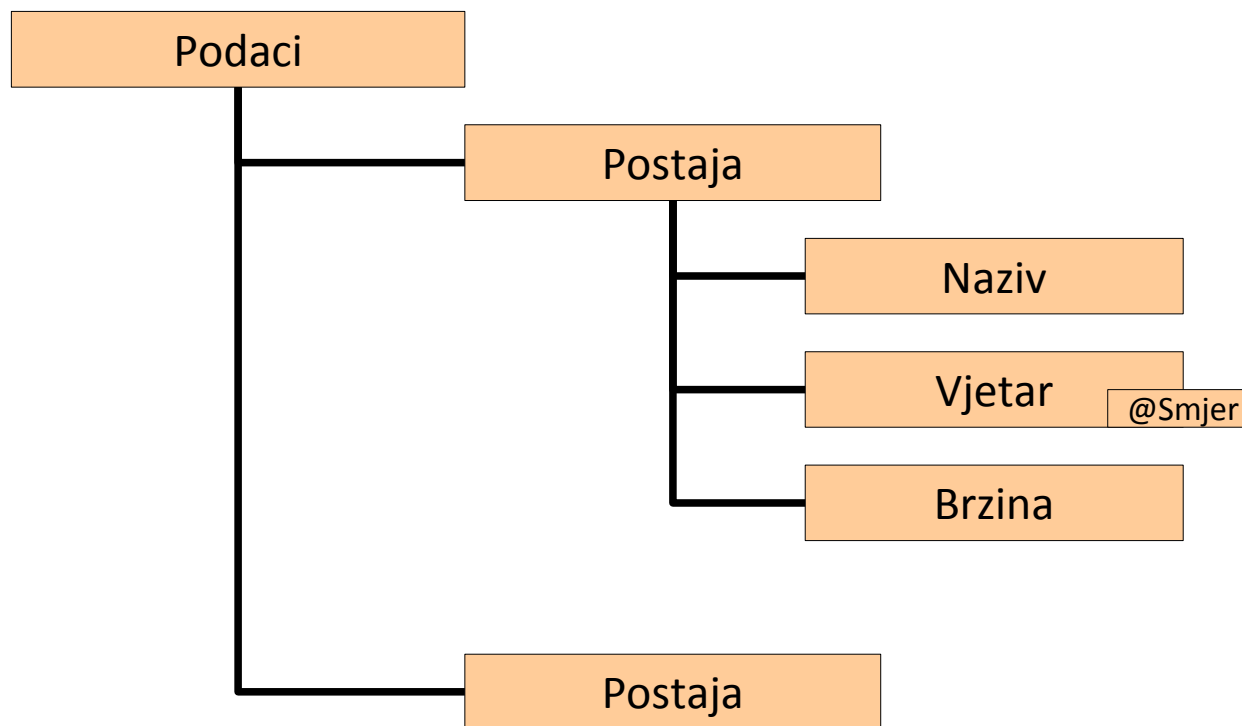
- U XML dokumentu treba navesti koji se XSL stil koristi

`<?xml-stylesheet type="text/xsl" href="transformacija.xsl"?>`

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!DOCTYPE podaci SYSTEM "vrijeme.dtd">
3 <?xml-stylesheet type="text/xsl" href="vrijeme.xsl"?>
```

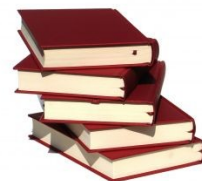

Način rada XSL

- CSS: selektor i promjena vrijednosti atributa
- XSL: traženje čvorova stabla (XPath) i zamjena sa predloškom



Trenutni kontekst

- Označava “gdje sam” trenutno
- Aktivni element u XPathu adresira korak
 - /root/.../predak/roditelj/**ČVOR**/dijete/potomak
- **ČVOR** je uvijek jednostruki čvor
 - može sadržavati samo jednog roditelja i jedan korijen
- Apsolutni put
 - /korak/korak/korak...
- Relativni put
 - korak/korak/korak...



Odabir čvor(ov)a



- Specijalni znakovi

/	odabir od korijenskog čvora
//	odabir od trenutnog čvora bez obzira na položaj
.	odabir trenutnog čvora
..	odabir roditelja trenutnog čvora
@	odabir atributa
*	bilo koji element čvora
@*	bilo koji atribut čvora
	odabir izraza prije ili iza ovog znaka

Primjer odabira čvorova



podaci	odabir sve djece elementa podaci
/podaci	odabir korijenskog elementa podaci
podaci/postaja	odabir svih postaja koji su djeca podataka
//postaja	odabir svih postaja bez obzira gdje su u dokumentu
./vjetar	odabir svih elementa vjetar, bilo gdje, ali potomaka trenutnog elementa
//@smjer	odabir svih atributa smjer bilo gdje u dokumentu
/podaci/*	odabir sve djece elementa podaci
//*	odabir svih elemenata u dokumentu



Predikati



- Filtar
- Služe za odabir čvora sa specifičnom vrijednosti
- Uokvireni uglatim zagradama – []
- Primjeri:
 - `postaja[1]`
 - `postaja[last()]`
 - `postaja[position()-1]`
 - `//vjetar[@smjer='N']`



Predlošci

- Predlošci se opisuju elementom *template*

`<xsl:template match="/"> ... </xsl:template>`

- Uvjet (*match*="...") je XPath izraz
 - Gornji primjer primjenjuje predložak na korijenski čvor
- Postoje i upravljačke strukture u XSL-u
 - Dohvaćanje sadržaja elementa/atributa
 - Iteratori/petlje (*for-each*)
 - Uvjetno izvođenje (*if/else, choose*)
 - Sortiranje (*sort*), ...
- Primjer

Predaja vježbe

- Poveznica u navigaciji na *podaci.xml*
- Prikazati **neke** podatke, da se stranica ne proširi
 - Ako bude "prozračno", tražit ćemo još :-P
- Prikazati barem jedan **podatak brojnosti N**
 - Odvojiti zarezom, napraviti mini-listu, staviti u novi red...
 - Barem 2x *for-each*
- Google Chrome!
 - Pokrenuti s parametrom: *--allow-file-access-from-files*

Rezultati vježbe

- Datoteke
 - podaci.xml
 - gramatika.dtd
 - pretvorba.xsl
- Ostale prateće datoteke (npr. CSS, slike)
- **VAŽNO!** Opet **predati strukturu podataka!**
- ORVViS vam može pomoći s provjerom ispravnosti predane vježbe :)

Poveznice

- XML Tutorial
 - www.w3schools.com/xml/
- DTD Tutorial
 - www.w3schools.com/dtd/
- XSL Tutorial
 - www.w3schools.com/xsl
 - www.nwalsh.com/docs/tutorials/xsl/



Hvala!

- 2. laboratorijska vježba
 - XML
 - DTD
 - XSL