

# Adreasá osob

**Dokumentace k zápo tovému p íkladu z IZI238**

**Michal Kec  
Vysoká škola ekonomická v Praze**

**E-mail: <xkecm01@vse.cz>**

**Web: <http://sorry.vse.cz/~xkecm01/>  
[<http://sorry.vse.cz/~xkecm01/>]**

**Adreasá osob: Dokumentace k zápo tovému p íkladu z IZI238**

Michal Kec

Copyright © 2001 Michal Kec

# Obsah

Úvod .....	iii
1. Popis a účel .....	1
1.1. Popis .....	1
1.2. Účel .....	3
2. XSLT, aneb z XML do HTML .....	4
3. XML až na papír .....	9
3.1. Z XML do FO .....	9
3.2. Z FO do PDF .....	14

# Úvod

Text, který právě začínáte číst je dokumentací k zápočtovému příkladu z předmětu *IZI238: XML -- teorie a praxe značkových jazyků*, vydaném od roku 2001 na Vysoké škole ekonomické v Praze [<http://www.vse.cz>].

Text byl napsán v textovém editoru programu FAR a v Notepadu. Celý dokument byl napsán a uložen ve formátu XML. S pomocí systému DocBook [<http://www.docbook.org>] a styl od Normana Walshe [<http://www.nwalsh.com>] lze zdrojový XML dokument snadno převést do různých formátů vhodných pro tisk, i publikaci na Internetu. Spolu se zdrojovým XML formátem je dokumentace k dispozici ve formátu HTML, RTF (ve verzi Word'95) a PDF.

Takto dokumentace, jakož i dokumentovaný zápočtový příklad je k dispozici na adrese [http://sorry.vse.cz/~xkecm01/p\\_links.php#izi238](http://sorry.vse.cz/~xkecm01/p_links.php#izi238)  
[[http://sorry.vse.cz/~xkecm01/p\\_links.php#izi238](http://sorry.vse.cz/~xkecm01/p_links.php#izi238)]



# Kapitola 1. Popis a ú el

## 1.1. Popis

Jednoduše e eno, jedná se o systém XML dokument a styl umož ůující vedení osobního, nebo firemního adresá e. Záznamem v adresá i je tedy jedna osoba a sledované údaje se d lí do n kolika skupin.

1. Osobní údaje
2. Bydlišt
3. Rodinné údaje
4. Zam stnání

Každá skupina dále obsahuje elementy specifické pro danou kategorii. Každá osoba musí mít vždy prá v jeden výskyt každé kategorie. Kategorie zam stnání je nepovinná. Podrobn jší informace o datové struktu e poskytne p iložení DTD.

```
<?xml encoding="windows-1250"?>
<!ELEMENT adresar (osoba+)>

<!ELEMENT osoba (osobni, bydliste, rodina, zamestnani?)>
<!ATTLIST osoba
  id ID #REQUIRED
  jmeno CDATA #REQUIRED
  prijmeni CDATA #REQUIRED
  titul_pred CDATA #IMPLIED
  titul_za CDATA #IMPLIED>

<!-- ===== -->
<!ELEMENT osobni (výška, váha, barva_o í?, barva_vlas ?, datum_narození,
<!ELEMENT výška (#PCDATA)>
<!ATTLIST výška jednotka CDATA 'cm'>
<!ELEMENT váha (#PCDATA)>
<!ATTLIST váha jednotka CDATA 'kg'>
<!ELEMENT barva_o í (#PCDATA)>
<!ELEMENT barva_vlas (#PCDATA)>
<!ELEMENT datum_narození (#PCDATA)>
<!ELEMENT rodné_íslo (#PCDATA)>
<!ELEMENT pohlaví (#PCDATA)>
<!ELEMENT ob anství (#PCDATA)>
```

```

<!ELEMENT národnost (#PCDATA)>

<!-- ===== -->
<!ELEMENT bydliste (okres?, m sto, ulice?, ps , telefon*, fax*, e-mail*)>

<!ELEMENT okres (#PCDATA)>
<!ELEMENT m sto (#PCDATA)>
<!ELEMENT ulice (#PCDATA)>
<!ELEMENT ps (#PCDATA)>
<!ELEMENT telefon (#PCDATA)>
<!ATTLIST telefon uto CDATA '02'>
<!ELEMENT fax (#PCDATA)>
<!ATTLIST fax uto CDATA '02'>
<!ELEMENT e-mail (#PCDATA)>

<!-- ===== -->
<!ELEMENT rodina (manžel-ka?, dítě *)>
<!ATTLIST rodina stav (svobodný|svobodná|žena|vdaná|vdovec|vdova|rozvedený|rozvedená) #REQUIRED>
<!ELEMENT manžel-ka (#PCDATA)>
<!ATTLIST manžel-ka
  id IDREF #IMPLIED
  jmeno CDATA #REQUIRED
  prijmeni CDATA #REQUIRED>
<!ELEMENT dítě (#PCDATA)>
<!ATTLIST dítě
  id ID #IMPLIED
  jmeno CDATA #REQUIRED
  prijmeni CDATA #REQUIRED>

<!-- ===== -->
<!ELEMENT zamestnani (název, funkce, m sto?, ulice?, ps ?, telefon*, fax*)>
<!ELEMENT název (#PCDATA)>
<!ELEMENT funkce (#PCDATA)>

```

Pro větší názornost přikládám i část zdrojového XML dokumentu se zvýrazněnou syntaxí tak, jak se zobrazuje v prohlížeči.

```

- <osoba id="N3" jmeno="Pavla" prijmeni="Nováková">
- <osobni>
  <výška>174</výška>
  <váha>68</váha>
  <barva_očí>hnědá</barva_očí>
  <barva_vlasů>hnědá</barva_vlasů>
  <datum_narození>14.3.1978</datum_narození>
  <rodné_číslo>785314/3252</rodné_číslo>
  <pohlaví>žena</pohlaví>
</osobni>
- <bydliste>
  <okres>Praha</okres>
  <město>Praha 10 - Strašnice</město>
  <ulice>Starostrašnická 35</ulice>
  <psč>10000</psč>
  <telefon>781 22 33</telefon>
  <e-mail>PNovak@email.cz</e-mail>
</bydliste>
- <rodina stav="vdaná">
  <manžel-ka id="N1" jmeno="Petr" prijmeni="Novák" />
  <dítě prijmeni="Novák" jmeno="Vašek" />
  <dítě prijmeni="Nováková" jmeno="Jana" />
</rodina>
- <zamestnani>
  <název>TV plus</název>
  <funkce>produkční</funkce>
  <město>Praha 1 - Staré město</město>
  <ulice>Václavské náměstí 38</ulice>
  <psč>10102</psč>
  <telefon>84 25 32 12</telefon>
  <fax>84 25 30 00</fax>
  <e-mail>Novakova@plus.tv</e-mail>
</zamestnani>
</osoba>

```

## 1.2. Ú el

Ú elem celé mé snahy bylo nau it se DTD, XML, jakož i XSLT a FO. Výsledné dílo je celkem použitelné, takže v budoucnosti a s drobným p izp sobením bych to mohl využít i pro osobní pot ebu. Motivací mi samoz ejm byla vidina zápo tu z p edm tu IZI238: XML -- teorie a praxe zna kovacíh jazyk . :-)

## Kapitola 2. XSLT, aneb z XML do HTML

Pro přehlednost jednotlivých záznamů jsem vybral formátování dat do tabulek. Každý záznam (každá osoba) má vlastní tabulku s vnitřním členěním podle čtyř hlavních kategorií. Abych se vyhnul problémům formátování v HTML, připojil jsem k HTML dokumentu list stylů (CSS), které obstarávají vizuální prezentaci (pozadí, barvy, ...)

Protože je dále přiložen vlastní XSL soubor, který je zde komentován, popíšu zde pouze stručně jeho obsah a detailněji rozvedu pouze některé části.

V první části se vyberou a podle abecedy seřadí všechny záznamy osob. Využívám zde toho, že jméno je uvedeno formou atributu `ko` nového elementu `osoba` a vypíšu celé jméno (i s tituly) jako záhlaví tabulky. Poté vygeneruji základní strukturu tabulky, do které se budou vkládat data z jednotlivých kategorií.

K osobním údajům se přidává také obrázek, který je pojmenován dle atributu `ID` (který je pro každou osobu jedinečný). Poté se postupně vypíší všechny osobní údaje včetně příslušných dodatečných údajů (jednotky).

Rodinné údaje: Pokud má manžel, i manželka právnické osoby vlastní záznam v adresáři (je uveden atribut `ID`), pak se vytvoří odkaz na vyhledání v seznamu. Část *D* *ti* se vypisuje pouze, pokud je v záznamu alespoň jedno dítě (využití funkce `<xsl:if>`).

Kategorie *Bydliště* a *Zaměstnání* mají podobnou strukturu, proto je popíšu jen stručně a dohromady. V pravé části tabulky se nachází adresa (u zaměstnání doplněná názvem firmy a funkcí). V pravé části pak jsou uvedeny telefony a faxy (včetně předvoleb) a e-maily formou odkazu. Kliknutím na odkaz se otevře asociovaný poštovní program

```
<?xml version="1.0" encoding="windows-1250"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
<xsl:output indent="yes" method="html" encoding="windows-1250"/>
```

```
<!-- Vytvoreni zakladni kostry dokumentu -->
<xsl:template match="/">
  <html>
    <head>
      <title>Adresář </title>
      <link type="text/css" rel="StyleSheet" title="Zakladni styly" href="adresar.css" />
    </head>
    <body>
      <xsl:for-each select="adresar/osoba">
        <!-- Seradime podle prijmeni -->
        <xsl:sort select="@prijmeni" />
        <xsl:sort select="@jmeno" />
```



```

<a name="{@id}"></a>
<!-- Kazdy zaznam ve vlastni tabulce. Zahlavi je cele jmeno -->
<table width="90%" border="1" align="center" cellpadding="3" cellspacing="3">
  <caption>
    <xsl:value-of select="concat(@titul_pred,' ',@jmeno,' ',@prijmeni,' ')" />
  </caption>
<xsl:apply-templates/>
</table>
</xsl:for-each>
</body>
</html>
</xsl:template>

```

```

<!-- Pro kazdy oddil samostatne zahlavi -->
<!-- Osobni udaje -->
<xsl:template match="osobni">
  <tr><th class="high">Osobní údaje</th>
  <td width="50%" rowspan="8" valign="center" align="center"></td>
</tr>
<!-- Vypis osobnich udaju. Kazdy na samostatne radce -->
<xsl:for-each select="*">
  <tr>
    <td>
      <b><xsl:value-of select="translate(name(),'_',' ')" />
      <xsl:text>: </xsl:text></b>
      <xsl:value-of select="." />
      <xsl:text> </xsl:text>
      <xsl:value-of select="@*" />
    </td>
  </tr>
</xsl:for-each>
</xsl:template>

```

```

<!-- Rodine udaje -->
<xsl:template match="rodina">
  <tr><th colspan="2" class="high">Rodiný stav: <xsl:value-of select="@stav" />
  <td colspan="2">
    <!-- Nejdriv testuje manzela/manzelku -->
    <xsl:apply-templates/>
    <!-- Potom testuje a vypisuje deti -->

```

```

<xsl:if test="dít ">
<tr><td><b>d ti:</b><ul>
<xsl:for-each select="dít ">
  <li>
    <xsl:value-of select="@jmeno" />
    <xsl:text> </xsl:text>
    <xsl:value-of select="@prijmeni" />
  </li>
</xsl:for-each>
</ul></td></tr>
</xsl:if>
</xsl:template>

```

```

<xsl:template match="manžel-ka">
<tr>
<td><b>manžel(ka):</b>
  <xsl:value-of select="@jmeno" />
  <xsl:text> </xsl:text>
  <xsl:value-of select="@prijmeni" />
  <!-- Pokud je maznel, ci manzelka v adresari, udelame nan odkaz -->
  <xsl:if test="@id"> <i>(<a href="#{@id}">Vyhledat</a></i></xsl:if>
</td>
</tr>
</xsl:template>

```

```

<!-- Bydliste -->
<xsl:template match="bydliste">
<tr><th colspan="2" class="high">Bydlišt </th></tr>
<!-- Do leve bunky adresu -->
<tr>
<td class="ods">
  <xsl:value-of select="ulice" /><br />
  <xsl:value-of select="ps " />
  <xsl:text>, </xsl:text>
  <xsl:value-of select="m sto" /><br />
  <i>(okres <xsl:value-of select="okres" />)</i>
</td>
<!-- Do prave bunky prijde to ostatni: tel, fax, mail -->
<td>
<xsl:apply-templates />
</td>
</tr>

```




```

<xsl:template match="text()">
  <!-- Sablona pro vynechani std. textu -->
</xsl:template>

</xsl:stylesheet>

```

Výsledek transformace do HTML si můžete prohlédnout na následujícím obrázku.

<b>Pavla Nováková</b>	
<b>Osobní údaje</b>	
<b>výška:</b> 174 cm	
<b>váha:</b> 68 kg	
<b>barva očí:</b> hnědá	
<b>barva vlasů:</b> hnědá	
<b>datum narození:</b> 14.3.1978	
<b>rodné číslo:</b> 785314/3252	
<b>pohlaví:</b> žena	
<b>Bydliště</b>	
Starostrašnická 35 10000, Praha 10 - Strašnice <i>(okres Praha)</i>	<b>telefon:</b> (02) 781 22 33 <b>e-mail:</b> <a href="mailto:PNovak@email.cz">PNovak@email.cz</a>
<b>Rodinný stav: vdaná</b>	
<b>manžel(ka):</b> Petr Novák ( <a href="#">Vyhledat</a> )	
<b>děti:</b>	
<ul style="list-style-type: none"> <li>• Vašek Novák</li> <li>• Jana Nováková</li> </ul>	
<b>Zaměstnání</b>	
<b>TV plus</b> produkční	<b>telefon:</b> (02) 84 25 32 12 <b>fax:</b> (02) 84 25 30 00 <b>e-mail:</b> <a href="mailto:Novakova@plus.tv">Novakova@plus.tv</a>
Václavské náměstí 38 10102, Praha 1 - Staré město	



```

<fo:block>
<xsl:for-each select="adresar/osoba">
<!-- Seradime podle prijmeni -->
<xsl:sort select="@prijmeni"/>
<xsl:sort select="@jmeno" />
<!-- Kazdy zaznam ve vlastni tabulce. Zahlavi je cele jmeno -->
<fo:block font="bold 200% Helvetica" break-before="page"
space-before="18pt" space-before.conditionality="discard"
space-after="6pt" keep-with-next.within-column="always"
keep-together.within-column="always" text-align="center"
padding="3pt" background-color="silver" id="{@id}">
<xsl:value-of select="concat(@titul_pred,' ',@jmeno,' ',@prijmeni,'" />
</fo:block>
<xsl:apply-templates/>
</xsl:for-each>
</fo:block>
</fo:flow>
</fo:page-sequence>

</fo:root>
</xsl:template>

<!-- Pro kazdy oddil samostatne zahlavi -->
<!-- Osobni udaje -->
<xsl:template match="osobni">
<fo:block font-family="Helvetica" font-size="150%" margin-top="12pt">Osc
<fo:table>
<fo:table-body>
<fo:table-cell starts-row="true">
<!-- Vypis osobnich udaju. Kazdy na samostatne radce -->
<xsl:for-each select="*">
<fo:block padding-left="0.5cm">
<fo:inline color="navy">
<xsl:value-of select="translate(name(),'_',' ')" />
<xsl:text>: </xsl:text>
</fo:inline>
<xsl:value-of select="." />
<xsl:text> </xsl:text>
<xsl:value-of select="@*" />
</fo:block>
</xsl:for-each>
</fo:table-cell>
<fo:table-cell ends-row="true">

```



```

    text-decoration="underline">manžel(ka):</fo:basic-link>
</xsl:when>
<xsl:otherwise>
  <!-- Jinak se napise jenom jmeno -->
  <fo:inline color="navy">manžel(ka):</fo:inline>
</xsl:otherwise>
</xsl:choose>
<xsl:text> </xsl:text>
<xsl:value-of select="@jmeno" />
<xsl:text> </xsl:text>
<xsl:value-of select="@prijmeni" />
</fo:block>
</xsl:template>

```

```

<!-- Bydliste -->
<xsl:template match="bydliste">
  <fo:block font-family="Helvetica" font-size="150%" margin-top="12pt">Byd
  <fo:table>
    <fo:table-body>
      <fo:table-cell starts-row="true">
        <!-- Do leve bunky adresu -->
        <fo:block padding-left="0.5cm">
          <xsl:value-of select="ulice" />
        </fo:block>
        <fo:block padding-left="0.5cm">
          <xsl:value-of select="ps " />
          <xsl:text>, </xsl:text>
          <xsl:value-of select="m sto" />
        </fo:block>
        <fo:block padding-left="0.5cm" font-size="85%">
          (okres <xsl:value-of select="okres" />)
        </fo:block>
      </fo:table-cell>
      <fo:table-cell ends-row="true">
        <!-- Do prave bunky prijde to ostatni: tel, fax, mail -->
        <fo:block padding-left="0.5cm">
          <xsl:text> </xsl:text>
        </fo:block>
        <xsl:apply-templates />
      </fo:table-cell>
    </fo:table-body>
  </fo:table>
</xsl:template>

```





```

        <xsl:value-of select="." />
    </fo:block>
</xsl:template>

<!-- Samostatna sablona vybira e-amil -->
<xsl:template match="e-mail">
    <fo:block padding-left="0.5cm">
        <fo:inline color="navy">
            e-mail:
        </fo:inline>
        <fo:basic-link external-destination="url(mailto:{.})"
            color="#0000C0" text-decoration="underline"><xsl:value-of select="." />
        </fo:block>
    </xsl:template>

<xsl:template match="text()">
    <!-- Sablona pro vynechani std. textu -->
</xsl:template>

</xsl:stylesheet>

```

## 3.2. Z FO do PDF

Poté, co získáme soubor s formátovacími objekty, můžeme tento libovolně upravovat, nebo (což je nepochybně větší zábava) z něj vygenerovat soubor vhodný pro tisk. Použil jsem omezenou verzi programu XEP [<http://www.renderx.com>] a vygeneroval výsledný PDF soubor.

Program XEP má bohužel problémy s češtinou, tudíž některé znaky prostě vynechává, což nepůsobí příliš dobře. Na druhou stranu XEP i v omezené demoverzi zvládá dobře standardy vznikající FO v etn tabulek, vícesloupcové sazby apod. Na rozdíl od plné verze demo neumí vytvořit PostScript verzi, do každé stránky vkládá odkaz na svou domovskou stránku a od jedenácté stránky je každá lichá stránka prázdná. :( Proto píšete nebudu hloupý a pojdím si zdarma a legálně najít jakýkoliv širší FO parser. Ale ta podpora!

Jak to vše dopadlo je zřejmé z následujícího obrázku.

