

E-kursu materiālu veidošanas un uzturēšanas metodes

Arnis Voitkāns

04.05.2010.

Anotācija

Izstrādājot e-kursus, parasti kursa materiālus nepietiek izveidot tikai vienā formā, jo neviens materiālu formāts nav vispiemērotākais visiem e-studiju pielietojumiem. Bieži e-kursa materiāli tiek izveidoti vairākās formās — parasti tie ir doc, html un pdf faili. Tomēr materiālus vairākās formās ir sarežģīti uzturēt, jo, veicot labojumus vienā dokumentā, šie paši labojumi ir jāveic arī pārējos dokumentos. Vai arī laika gaitā ir jāatsakās no kādas no materiālu formas.

Paliek neatbildēts jautājums — kurš no failu formātiem tad ir piemērotākais e-kursu materiālu izstrādē? Kurā no formātiem ir vērts ieguldīt resursus — laiku un naudu, izstrādājot e-kursu materiālus?

Šajā rakstā ir apkopotas pārdomas par iespējām atvieglot materiālu vairākās formās izstrādi un, galvenais, uzturēšanu. Tiek sniegtas e-kursa materiālu avota failu formāta vispārīgās īpašības, kā arī analizēti \LaTeX un DocBook formāti.

Saturs

1	Ievads	2
2	E-kursa materiālu avota failu formāta vispārīgās īpašības	3
3	Datubāžu izmantošana e-kursu materiālu izveidē	3
4	\LaTeX kā avota dokumentu formāts	5
4.1	Īsumā par \LaTeX	5
4.2	$X_{\text{Y}}\LaTeX$ un $X_{\text{Z}}\LaTeX$	6
4.3	$X_{\text{Y}}\LaTeX$ dokumenta piemērs	7
4.4	\LaTeX dokumentu pārveide citos formātos	8
4.5	Priekšrocības un trūkumi	9
5	DocBook kā avota dokumentu formāts	10
5.1	Kas ir DocBook	10
5.2	DocBook pielietojums e-mācībās	10
5.3	DocBook dokumentu procesēšanas shēma	11
5.3.1	DocBook rediģēšana	11
5.3.2	FO vai \LaTeX ?	13
5.3.3	PDF un HTML dokumentu formatēšana	14
5.4	DocBook piemērs	15
5.5	Matemātika DocBook un LaTeX dokumentos	16
5.5.1	MathML izmantošanas piemērs	17
5.6	DocBook priekšrocības un trūkumi	18
6	Nobeigums	19

Literatūra	19
A DocBook fails	21
B XSLT transformāciju fails	22
C Generētais \LaTeX fails	23

Attēlu saraksts

1	Principiālā shēma	4
2	Ģenerētais PDF dokuments	8
3	DocBook procesēšanas shēma [20]	12
4	Vienkāršota DocBook procesēšanas shēma, izmantojot \LaTeX	14
5	XMLmind WYSIWYG DocBook redaktora ekrānkopija	15
6	Izveidotais DocBook piemēra PDF dokuments	16
7	Formula WebEQ redaktorā	18

1 Ievads

Ļoti svarīga e-kursa sastāvdaļa ir kvalitatīvi e-kursa materiāli¹. E-kursa izstrādes laikā materiālus vairākās formās ir iespējams izveidot bez īpašām problēmām. Piemēram, par pamatdokumentiem var izmantot Word dokumentus, no kuriem beigās iespējams izveidot PDF un HTML failus, kā arī nepieciešamības gadījumā prezentāciju dokumentus. Tomēr no Word vai OpenOffice dokumentiem izveidot kvalitatīvus HTML dokumentus² var būt sarežģīti un darbietilpīgi. Jo materiālu avota Word dokumenti kļūst sarežģītāki, jo grūtāk izveidot HTML failus³. Konvertēt Word vai OpenOffice dokumentus uz PDF formātu ir salīdzinoši vienkārši.

Šādas materiālu veidošanas galvenais trūkums ir tāds, ka vēlāk materiālus ir arvien sarežģītāk uzturēt — ir sarežģīti veikt izmaiņas un nodrošināt, lai izmaiņas tiktu korekti veiktas visās materiālu formās. Parasti kļūdas ir jālabo visās trijās materiālu formās — sākumā Word vai OpenOffice dokumentos, tad no jauna ir jāģenerē atbilstošie PDF dokumenti, un šie paši labojumi ir jāveic arī HTML dokumentos (jo no jauna izveidot HTML dokumentus no Word dokumentu oriģināliem ir pārāk darbietilpīgi).

Lai saprastu, kā optimāli veidot e-kursu materiālus, sākumā noskaidrosim, kādām jābūt materiālu avota failu⁴ formāta vēlamajām īpašībām. Tad apskatīsim vairākus iespējamus avota failu formātus, un izanalizēsim to atbilstību vispārīgajām īpašībām. Galvenais uzdevums ir apskatīt tikai tādus avota failu formātus, kas sniedz iespēju izveidot materiālus vairākās formās, kuras pēc tam ir ērti atjaunot un uzturēt.

¹Ar kvalitatīviem materiāliem šeit tiek saprasti: 1) materiāli bez gramatiskām un saturiskām kļūdām, 2) aktuāli materiāli (tajos nav novecojušas informācijas), 3) materiāli ir ērti izmantojami tiešsaistē un ir pieejama kvalitatīva izdrukas versija. Lai to varētu nodrošināt, materiāliem ir jābūt pieejamiem gan HTML, gan arī PDF formātos.

²Ar kvalitatīviem HTML dokumentiem šeit tiek saprasti dokumenti, kuri 1) nesatur liekus tagus, 2) kuriem ir optimāla iekšējā struktūra, 3) tiem ir mazs apjoms kilobaitos, līdz ar to tie ātrāk lejupielādējas, 4) dokumentiem ir laba pieejamība — tie ir korekti apskatāmi ar lielāku pārlūkprogrammu skaitu.

³Grūtības ar Word vai OpenOffice dokumentu pārveidi citos formātos var būt saistītas arī ar to, ka Word vai OpenOffice dokumentu autori bieži formatē tikai dokumentu izskatu un neraksta tos strukturētā veidā. Protams, šos dokumentus ir iespējams rakstīt strukturētā veidā, tomēr tik un tā tos konvertēt par labiem HTML dokumentiem var būt sarežģīti, jo bieži konvertētais HTML kods vēl ir jāattīra no liekiem tagiem un tad papildus jāformatē.

⁴Šeit un turpmāk tekstā ar avota failiem tiek saprasti e-kursa materiālu pamatdokumenti, no kuriem pēc tam tiek ģenerētas sekundārās materiālu formas — HTML un PDF dokumenti.

Šajā rakstā tiek apskatīti trīs veidi, kā organizēt e-kursa materiālu izstrādi:

- Datubāzu izmantošana e-kursa materiālu glabāšanā;
- \LaTeX kā avota dokumentu formāts;
- DocBook kā avota dokumentu formāts.

2 E-kursa materiālu avota failu formāta vispārīgās īpašības

Uzskaitīsim svarīgākās īpašības, kurām jāpiemīt e-kursa avota failu formātam. Īpašības ir apkopotas, balstoties uz autora daudzu gadu pieredzi, strādājot ar e-kursu materiāliem.

- Vienā un tajā pašā dokumentā, nepārslēdzot kodējumus, ir jāvar rakstīt tekstu vairākās valodās.

Tā ir obligāta prasība. Piemēram, pamatteksts latviešu valodā, citāti krievu vai citās valodās, literatūras sarakstā grāmatu nosaukumi krievu, vācu un c. valodās.

Parasti tas nozīmē, ka dokumentu formātam iekšēji ir jāatbalsta UTF-8 (vai Unicode) kodējums;

- Visas nacionālās rakstzīmes avota tekstā ir jāvar rakstīt no klaviatūras tieši (neizmantojot to kodus), lai būtu ērta teksta ievade un laba lasāmība;

- Avota failu formātam ir jābūt pēc iespējas neatkarīgākam no datoru arhitektūrām un operētājsistēmām. Tādā veidā tiek nodrošināta avotu failu plaša pieejamība visdažādākajās datoru konfigurācijās.

Jāpiezīmē, ka atklātā teksta faili⁵ UTF-8 kodējumā ideāli atbilst šim kritērijam;

- Formātam ir jāatbalsta un jāveicina strukturētu dokumentu izstrāde;
- Saturam ir jābūt nodalītam no pasniegšanas formas;
- No avota dokumentiem ir jāspēj ērti ģenerēt gan HTML dokumentus UTF-8 kodējumā, gan arī drukāšanas materiālus PDF formātā;
- Vēlams, lai būtu iespējams automātiski ģenerēt kvalitatīvus PDF dokumentus ar iekšējām saitēm un grāmatzīmēm (*bookmarks*).

3 Datubāžu izmantošana e-kursu materiālu izveidē

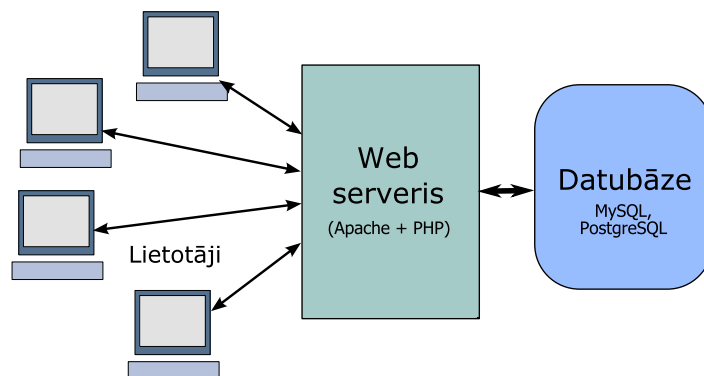
E-kursa materiālu glabāšana ir iespējama izmantot datubāzes. Kopumā šī tehnoloģija ir līdzīga metodēm, ko izmanto daudzās tīmekļa vietnēs.

Parasti šādās sistēmās ir divas galvenās komponentes — tīmekļa serveris un datubāze. Datubāzes galvenā funkcija ir e-kursu datu glabāšana, savukārt tīmekļa servera uzdevums ir atlasīt datus no datubāzes, tos apstrādāt un pasniegt lietotājam.

Kā tīmekļa serveri var izmantot *Apache*⁶. *Apache* tīmekļa serveris ir atklātā pirmkoda produkts, un tas ir ļoti populārs. Tomēr bez tīmekļa servera vēl ir nepieciešama tehnoloģija, kas veiktu datu atlasīšanu no datubāzes un gatavotu dinamiskas lapas. Ļoti populāra

⁵angl. *plaintext*

⁶<http://www.apache.org/>



Att. 1: Principiālā shēma

un jaudīga bezmaksas tehnoloģija, kuru izmanto daudzu dinamisku tīmekļa lapu veidošanā, ir PHP⁷. Protams, tā nav vienīgā tehnoloģija, ir arī citas, piemēram, ASP vai JSP. Kā datubāzi var izmantot *MySQL*⁸ vai *PostgreSQL*⁹. Principiālā šādas konfigurācijas shēma parādīta 1. attēlā.

Priekšrocības:

- ✓ Ir iespēja ģenerēt dinamiskus materiālus — kad lietotājs ver vaļā materiālus, pieprasītie dokumenti katru reizi tiek no jauna ģenerēti, izmantojot datus no datubāzes;
- ✓ Šī metode ir samērā ātra — *MySQL* datubāze un *PHP* ir optimizēti darbam internetā. Kombinācijā ar kešošanas infrastruktūru tādi tīmekļa serveri var izturēt lielas darba slodzes. Tomēr jāpiebilst, ka e-mācību jomā materiālu ģenerācijas ātrumam nav izšķiroša nozīme, jo tie nav jāatjauno ļoti bieži. Kad materiālos ir veiktas izmaiņas, pietiek tos tad atjaunot;
- ✓ *PHP* atļauj dinamiski ģenerēt PDF dokumentus un attēlus.

Trūkumi:

- ✗ Ir nepieciešams serveris vai vairāki datubāzei un tīmekļa serverim;
- ✗ Lai strādātu ar serveri, ir nepieciešams interneta pieslēgums. Var būt grūtības veidot materiālus, ja nav pieejams internets;
- ✗ Serveris ir jāuztur. Ja serveris nedarbosies, lietotāji netiks klāt pie e-studiju materiāliem;
- ✗ Kopumā, ja nepieciešams izstrādāt lietotāju saskarni datu ievadei, šīs metodes realizācija ir sarežģītāka par pārējām zemāk aprakstītajām metodēm.

Tomēr mūsdienās internetā ir pieejamas daudzas jau gatavas sistēmas, arī atklātā pirmkoda, kuras var izmantot, lai izstrādātu un glabātu materiālus tīmeklī. Tās ir satura pārvaldības sistēmas (piemēram, *Typo3*¹⁰, *Joomla*¹¹, *Drupal*¹²), blogu sistēmas (piemēram, *WordPress*¹³), viki sistēmas (piemēram, *MediaWiki*¹⁴) un e-izglītības vides (Latvijā

⁷<http://www.php.net/>

⁸<http://www.mysql.com/>

⁹<http://www.postgresql.org/>

¹⁰<http://typo3.org/>

¹¹<http://www.joomla.org/>

¹²<http://drupal.org/>

¹³<http://lv.wordpress.org/>

¹⁴<http://www.mediawiki.org/wiki/MediaWiki>

universitāšu vidē vispopulārākā ir Moodle¹⁵). Gatavo sistēmu izmantošana ievērojami atvieglo un paātrina šīs metodes realizāciju. Tomēr minētās sistēmas ir vairāk paredzētas tīmekļa dokumentu (lapu) izstrādei, glabāšanai un uzturēšanai, un mazāk tipogrāfiskas kvalitātes drukas materiālu izveidei.

Zemāk aprakstītā \LaTeX sistēma tiek izmantota kāreiz tipogrāfiskas kvalitātes drukas materiālu izveidei.

4 \LaTeX kā avota dokumentu formāts

4.1 Īsumā par \LaTeX

\LaTeX ir dokumentu sagatavošanas sistēma, kas balstīta uz \TeX [5].

\TeX ir tipogrāfisko dokumentu salikšanas programma, kurai jau sākotnēji tika izvirzīti divi uzdevumi: [6]

- Nodrošināt katram iespēju veidot augstas kvalitātes publikācijas un grāmatas, pieliekot saprātīgu piepūļu daudzumu;
- Sniegt sistēmu, kas nodrošinātu identiskus rezultātus uz visiem datoriem, tagad un nākotnē.

\TeX ir uzrakstījis prof. Donalds Knuts (grāmatas “The art of computer programming” autors¹⁶). Tā kā prof. Knuts nebija apmierināts ar savas grāmatas izskatu, viņš 1970'to gadu beigās uzsāka rakstīt \TeX . Lai gan prof. Knuts bija ieplānojis, ka \TeX izstrāde viņam prasīs 6 mēnešus, viņam bija nepieciešami gandrīz 10 gadi, lai programmu pabeigtu [9].

Darbošanās ar \TeX atgādina programmēšanu. Būtībā \TeX arī ir programmēšanas valoda, kas orientēta uz tipogrāfisko salikšanu. Ir iespējams deklarēt savas teksta noformēšanas komandas un makrosus. Tomēr darbošanās ar \TeX sistēmu nepasatārināti nav pārāk ērta. Šī iemesla dēļ, lai atvieglotu un automatizētu dokumentu sagatavošanu, tika izstrādāts \LaTeX . Jāpiebilst, ka šobrīd pasaulē \LaTeX ir kļuvis par dominējošo veidu, kā strādāt ar \TeX tipogrāfisko programmu.

\LaTeX automatizē lielāko daļu no dokumentu sagatavošanas darbībām, piemēram, numerācijas, atsauču, satura rādītāja un indeksu izveidi. Tā pamatā ir ideja, ka autoriem būtu jākoncentrējas uz dokumentu loģiskās struktūras un satura izstrādi, nevis jāvelta savs laiks formatēšanai. Rakstot \LaTeX dokumentus, ir iespējams nodalīt saturu no formas. Šī iemesla dēļ \LaTeX bieži tiek uzskatīts par pārāku par parastajiem teksta procesoriem, kuri parasti sasaista saturu un formu tik cieši, ka šādus dokumentus vēlāk ir grūti transformēt un nodrošināt to saskanīgumu [5].

\LaTeX dokumentus ir iespējams samērā vienkārši pielāgot. Ja nepieciešams, var pārdefinēt \LaTeX komandas, lai iegūtu tieši tādu dokumenta izskatu, kādu autors vēlas. Ir iespējams arī izstrādāt savus dokumentu stilus. Pie tam, tā kā \LaTeX dokumentu saturs ir nodalīts no formas, lai būtiski izmainītu dokumenta noformējumu, nav nepieciešams mainīt saturu.

Jāpiebilst, ka jau no pašiem pirmsākumiem \TeX tika orientēts uz dokumentiem ar sarežģītu matemātiku. Tādēļ \LaTeX dokumentu sagatavošanas sistēma, jo tā ir balstīta uz \TeX , sniedz ārkārtīgi labu atbalstu matemātikas dokumentu rakstīšanai. Vēl joprojām daudzi matemātikas un fizikas žurnāli pieņem publikācijas \LaTeX formātā, un daudzi matemātiķi neredz alternatīvu \LaTeX , ja ir runa par matemātikas publikāciju un žurnālu izdošanu.

¹⁵<http://moodle.org/>

¹⁶<http://www-cs-faculty.stanford.edu/~knuth/taocp.html>

4.2 XeTeX un XeLaTeX

Sakarā ar to, ka TeX ir izstādāts pagājušajā gadsimta 70-tajos un 80-tajos gados, kad vēl nebija pazīstams UTF-8 kodējums¹⁷, tradicionālajām TeX un LaTeX sistēmām ir vairāki ļoti būtiski trūkumi:

- ✗ Standarta LaTeX sistēmās iekšēji tiek izmantoti 8-bitu fonti — tas nozīmē, ka fontā nevar būt vairāk par 256 rakstzīmēm. Šis ir ļoti būtisks ierobežojums, jo tādā gadījumā ar vienu LaTeX fontu vienā kodējumā vienlaicīgi nevar attēlot daudzu valodu rakstzīmes. Tas arī nozīmē, ka LaTeX tiešā veidā neprot strādāt ar Unikoda fontiem.

Tomēr arī standarta LaTeX sistēmās šo ierobežojumu ir iespējams apiet, izmantojot avota dokumenta kodējumu (*inputencoding*) UTF-8, un dokumentā pārslēdzot fonta kodējumu (*fontencoding*) uz vajadzīgo LaTeX 8-bitu kodējumu. Šādā veidā vienā dokumentā ir iespējams rakstīt gan latviešu, gan arī krievu valodās, tomēr šis paņēmieni kopumā ir samērā neērta un ne vienmēr dod labu rezultātu.

- ✗ Latviešu valodai LaTeX'ā nebija publiski pieejamu TrueType fontu, kas spētu strādāt ar tradicionālo TeX dzinēju. Tomēr TrueType fontu atbalsts latviešu valodai standarta sistēmā ir iespējams, raksta autoram to izdevās panākt. Autoram izdevās pielāgot esošos TrueType fontus (piemēram, Palatino Linotype, Times New Roman, Arial u.c.), lai tos varētu izmantot LaTeX dokumentos. Tomēr ne visus fontus, kurus raksta autors bija mēģinājis pielāgot, izdevās piemērot LaTeXam;

- ✗ Nevienā standarta LaTeX instalācijā pēc noklusējuma nebija korekta latviešu valodas atbalsta. Latviešu valodas atbalsts bija jāliek lietotājiem pašiem. Arī pieejamais LaTeX'a latviešu valodas atbalsts bija nepilnīgs. Latviešu valodas atbalstam tradicionālajās LaTeX sistēmās ir veltīta Ivāra Driķa interneta vietne [12], kurā ir LaTeX latviskošanas iespēju apraksts, kā arī ir iespējams lejupielādēt datus ar latviskošanu saistītus failus.

I.Driķa lapā ir latviešu failu atbalsts bija tīja CP1257 kodējuma failiem, kas ir 8-bitu kodējums un pilnīgi neder e-kursu avota materiālu formātam. Tomēr, kā jau tika minēts iepriekš, raksta autoram izdevās izveidot ierobežotu latviešu valodas UTF-8 atbalstu tradicionālajām LaTeX sistēmām.

Lai apietu būtiskos TeX sistēmas ierobežojumus, relatīvi neseno¹⁸ tika izveidots modificēts TeX dzinējs, kuru sauc par XeTeX (vai vienkāršā tekstā XeTeX). XeTeX ir TeX tipogrāfiskās salikšanas dzinējs, kas atbalsta Unikodu (*Unicode*) un modernas fontu tehnoloģijas, tādas kā OpenType vai AAT (*Apple Advanced Typography*). XeTeX var izmantot jebkādas fontus, kas ir instalēti datora operētājsistēmām, un var izmantot papildus fontu tipogrāfiskās iespējas. XeTeX darbojas gan Linux, gan Mac OS X, gan arī Windows vidēs. [8, 7]

XeLaTeX atbalsta lielāko daļu no LaTeX rakstītajām makrosu pakotnēm. Lai nokompilētu LaTeX dokumentus ar XeTeX, ir jāizmanto komanda `xelatex`.

Modernā XeTeX sistēma novērta praktiski visus trūkumus, kas ir minēti šajā nodaļā. Tāpēc novecojušos LaTeX latviešu valodas atbalsta failus var vairs nelietot, un turpmāk šajā rakstā tiks apskatīts tieši XeLaTeX lietojums, tā priekšrocības un trūkumi (kas ir arī LaTeX priekšrocības un trūkumi). Lietojot teminu LaTeX, ar to tiks saprasts LaTeX lietojums XeTeX sistēmā.

¹⁷UTF-8 kodējums oficiāli tika prezentēts tikai 1993. gadā. [4]

¹⁸Pirmā publiski pieejamā XeTeX versija tika izlaista 2004. gadā un tikai Mac OS X operētājsistēmai. Pirmā XeTeX versija ar pilnu latviešu valodas atbalstu (versija 0.999.6), kas tika iekļauta TeX Live 2008 distributīvā, tika izlaista tikai 2008. gada septembrī.

4.3 \LaTeX dokumenta piemērs

Šeit ir redzams vienkāršs \LaTeX dokumenta piemērs:

```
1 \documentclass[a4paper,11pt]{article}
2
3 % XeLaTeX atbalsts tiek pieslēgts ar šādām pakotnēm:
4 \usepackage{fontspec}
5 \usepackage{xunicode}
6 \usepackage{xltextra} %
7
8 % Valodu atbalsts
9 \usepackage{polyglossia}
10 \setdefaultlanguage{latvian}
11 \setotherlanguages{english,russian}
12
13 % Fonti -- var rakstīt sistēmas fontu nosaukumus
14 \setmainfont[Mapping=tex-text]{Times New Roman}
15 \setsansfont[Mapping=tex-text]{Arial}
16 \newfontfamily\russianfont{Times New Roman}
17
18 % =====
19
20 \begin{document}
21
22 \title{Šis ir \LaTeX dokumenta piemērs}
23 \author{Autors}
24 \maketitle
25
26 \begin{abstract}
27 Šeit ir dokumenta anotācija... Anotācija      automātiski      tiek      noformēta
28 ar mazāku fonta izmēru.
29 \end{abstract}
30
31 \section{Pirmā tēma}
32
33 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts
34 latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts latviešu
35 valodā... Šis ir teksts latviešu valodā...
36
37 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts
38 latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts latviešu
39 valodā...
40
41 \[E=mc^2\]
42
43 \[-\frac{\partial^2\psi_1(x)}{\partial x^2}\cdot\frac{1}{\psi_1(x)}-
44 \frac{\partial^2\psi_2(y)}{\partial y^2}\cdot\frac{1}{\psi_2(y)}-
45 \frac{\partial^2\psi_3(z)}{\partial z^2}\cdot\frac{1}{\psi_3(z)}=k^2\]
46
47 \section{Otrā tēma}
48
49 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts
50 latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts latviešu
51 valodā... Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis
52 ir teksts latviešu valodā... Šis ir teksts latviešu valodā...
53
54 \end{document}
```

1. rindā tiek pateikts, ka šis dokuments ir raksts (article), lapas izmērs ir A4, un fonta izmērs ir 11pt.

4.–6. rindās tiek pieslēgtas \LaTeX nepieciešamās pakotnes. *UTF-8* atbalsts šeit ir jau pēc noklusējuma.

9.–11. rindās tiek pieslēgtas pakotnes, kas paredzētas \LaTeX valodu atbalstam. Kā noklusētā valoda tiek uzlikta latviešu, kā alternatīvās valodas tiek pieslēgtas angļu un krievu.

13.–16. rindās nav obligāti nepieciešamas. Šajās rindās tiek pateikts, kādus fontus lietot. Ja nekādi fonti norādīti netiek, tiek izmantoti noklusētie \LaTeX fonti. Šajā piemērā ir ierakstīti plaši pazīstamie fonti “Times New Roman” un “Arial”.

Avota dokumenta failā starp vārdiem un teikumiem var būt pēc patikas daudz atstarpu. \LaTeX tās ignorē un atstāj tikai vienu (tas ir līdzīgi arī HTML kodam — pārlūkprog-

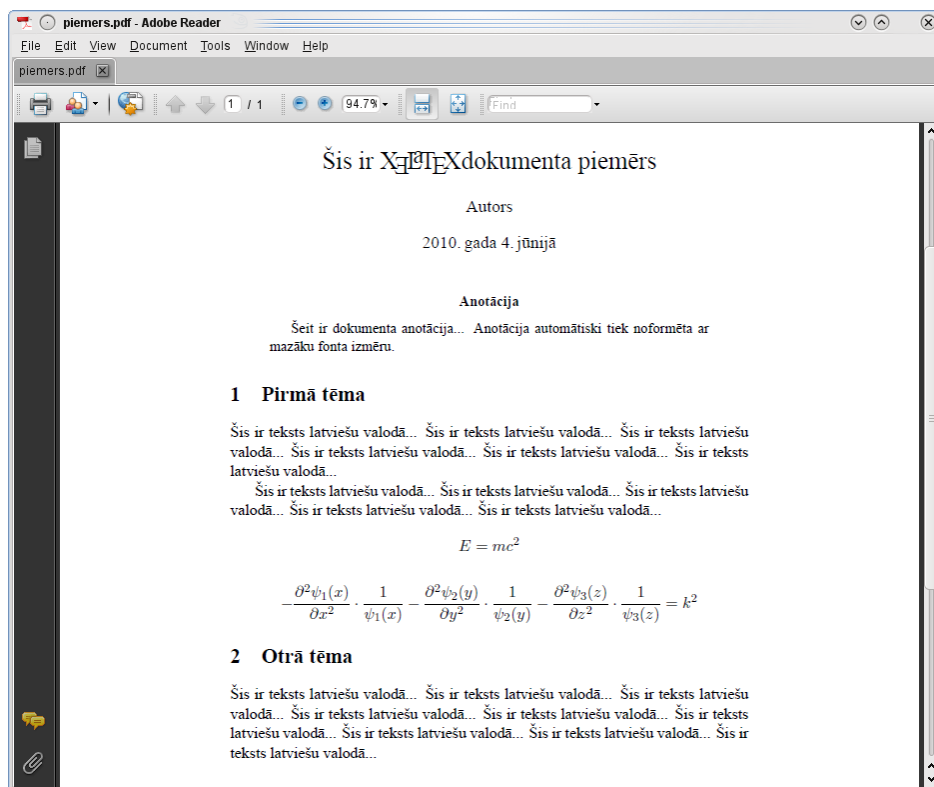
rammas ignorē liekās atstarpes). Jauna rindkopa tiek iesākta, ieliekot starp rindkopām vienu vai vairākas tukšas rindas.

Nodaļu un apakšnodaļu, tāpat arī tabulu un attēlu numerācija tiek veikta automātiski. *41. rindā un 43.–45. rindās* ir uzrakstītas divas formulas.

Lai no \LaTeX faila ģenerētu PDF dokumentu, izmantojot $X\TeX$, jālieto komanda `xelatex`. Šajā gadījumā tas izskatās tā:

```
1 > xelatex piemers.tex
```

Rezultējošais PDF dokuments redzams 2. attēlā.



Att. 2: Ģenerētais PDF dokuments

Lai tuvāk iepazītos ar \LaTeX , var ieteikt grāmatas “A beginner’s introduction to typesetting with \LaTeX ” [1] un “A (Not So) Short Introduction to $\LaTeX 2\epsilon$ ” [2]. Šīs grāmatas ir par \LaTeX , bet tajās aprakstītās struktūras pamatā der arī $X\TeX$ sistēmai.

4.4 \LaTeX dokumentu pārveide citos formātos

\LaTeX faili ir ideāli piemēroti dokumentu, kas paredzēti drukāšanai, izveidei. No \LaTeX failiem ir iespējams viegli ģenerēt ļoti augstas kvalitātes PDF dokumentus, un, jāpiezīmē, ka \LaTeX ir viena no retajām programmām, kas spēj ģenerēt dokumentus drukāšanai no avota dokumentiem atklātā teksta formātā. Tomēr paliek neatbildēts jautājums — kā var konvertēt \LaTeX dokumentus uz citiem formātiem, piemēram, HTML.

Tā kā \LaTeX dokumenti ir atklātā teksta faili, tad tos būtu jāvar samērā viegli konvertēt uz citiem formātiem. Tomēr tas ne vienmēr ir tā. Vienkāršus \LaTeX dokumentus konvertēt uz citiem formātiem ir viegli, bet problēmas sākas tad, kad \LaTeX dokumenti kļūst sarežģītāki. Izrādās, ka sarežģītus \LaTeX dokumentus korekti spēj apstrādāt tikai pats \LaTeX (vai

X₃TeX). L^AT_EX dokumentu autoriem ir iespējams pārdefinēt L^AT_EX komandas, veidot sarežģītus, savām vajadzībām pielāgotus, stilus. Konvertācijas programmas ar tādiem dokumentiem bieži nespēj tikt galā. Šī iemesla dēļ daudzi autori un izdevēji raksta paši savas konvertācijas programmas. Parasti tiek izmantotas skriptu valodas, piemēram, Perl. [1]

Eksistē arī gatavas L^AT_EX konvertācijas programmas. Var pieminēt divas plašāk pazīstamās — L^AT_EX2HTML [10] un T_EX4ht [11]. T_EX4ht prot konvertēt L^AT_EX dokumentus uz daudziem formātiem, tai skaitā HTML un XML. Jāpiezīmē, ka T_EX4ht spēj konvertēt L^AT_EX matemātikas formulas uz MathML formātu. T_EX4ht ir arī X₃TeX atbalsts.

Diemžēl, raksta autoram bija grūtības izmantot T_EX4ht gadījumos, kad L^AT_EX dokumenti ir UTF-8 kodējumā un satur latviešu zīmes.

4.5 Priekšrocības un trūkumi

Apkoposim L^AT_EX kā e-studiju materiālu avota dokumentu formāta priekšrocības un trūkumus.

Priekšrocības:

- ✓ Pārnesamība. L^AT_EX dokumentus iespējams “kompilēt” daudzās operētājsistēmās — Windows, Linux, Mac OS X u.c.;
- ✓ L^AT_EX ir pieejams bez maksas. Protams, pastāv arī maksas L^AT_EX distributīvi, bet standarta L^AT_EX sistēmas ir brīvi pieejamas;
- ✓ L^AT_EX dokumenti ir atklātā teksta faili, tādēļ tos iespējams rediģēt ar jebkuru teksta redaktoru. Protams, ir ieteicams izmantot specializētus redaktorus;
- ✓ No L^AT_EX dokumentiem ir iespējams iegūt tipogrāfiskas kvalitātes PostScript un PDF dokumentus. Ir iespējams automātiski ģenerēt PDF dokumentus ar iekšējām saitēm un grāmatzīmēm;
- ✓ L^AT_EX'ā ir ļoti labs matemātikas atblasts, kas ļauj rakstīt dokumentus ar sarežģītu matemātiku;
- ✓ L^AT_EX dokumentos ir iespējams nodalīt saturu no formas. Tādā veidā ir iespējams standartizēt dokumentu pasniegšanas stilus;
- ✓ L^AT_EX dokumentus gandrīz nav iespējams rakstīt nestrukturētus (vismaz tas ir ļoti neērti);
- ✓ L^AT_EX dokumenti tiek procesēti no komandrindas. Tas ļauj apstrādāt dokumentus ar skriptiem;
- ✓ L^AT_EX ir atvērta un viegli paplašināma sistēma — var veidot savus fontu kodējumus, valodu atbalstus, komandas un c.;
- ✓ Modernos T_EX distributīvos ir iekļauta X₃TeX sistēma, kura atbalsta L^AT_EX dokumentu rakstīšanu UTF-8 kodējumā, un kurai pēc noklusējuma ir korekts latviešu valodas atbalsts (to ir iespējams relatīvi vienkārši ieslēgt). Tomēr latviešu valodas atbalsts ir nepilnīgs.

Trūkumi:

- ✗ Ir problēmas ar standarta rīkiem konvertēt L^AT_EX dokumentus, kuri ir UTF-8 kodējumā. Ar X₃L^AT_EX parādīšanos, šajā aspektā ir vērojams progress;
- ✗ Nepietiekami elastīgs tabulu atbalsts, kas rada grūtības dokumentu ar lielām un sarežģītām tabulām izveidē;
- ✗ Bibliogrāfisko atsauču programmai BibTeX ir problēmas ar latviešu valodas atbalstu;
- ✗ Nepilnīgs latviešu valodas atbalsts.

Noklusētā \LaTeX sistēma neatbilst visām prasībām, kas ir formulētas 2. nodaļā. Taču, ja pieliek papildus piepūles (uzstāda un izmanto X_{\LaTeX} sistēmu), \LaTeX ir iespējams pielāgot galvenajām avota failu formāta prasībām, un \LaTeX var tikt izmantots kā e-studiju materiālu avota formāts. Tikai šajā gadījumā ir grūtības ģenerēt HTML failus.

Lai arī ir grūtības ar \LaTeX dokumentu konvertāciju uz citiem formātiem, kas īpaši izpaužas tad, ja dokumenti ir latviešu valodā, \LaTeX (vai, precīzāk, X_{\LaTeX}) ir ļoti piemērots kvalitatīvu PDF dokumentu ģenerācijai. Tomēr lietderīgāk e-studiju materiālu avota dokumentus turēt kādā neitrālā formātā, piemēram, uz XML bāzētos dokumentos, un tad no šiem avota dokumentiem ģenerēt \LaTeX failus un pēc tam PDF dokumentus. Ja tiek izmantoti XML faili, tas paver vēl papildus iespējas. Sikāk tas tiek apskatīts nodaļā par DocBook (skat. 5. nodaļu).

5 DocBook kā avota dokumentu formāts

5.1 Kas ir DocBook

DocBook ir XML/SGML valodas paveids, kas domāts rakstu un grāmatu rakstīšanai. Precīzāk sakot, DocBook ir XML vai SGML DTD¹⁹, kuru uztur OASIS²⁰ DocBook Tehniskā komisija. OASIS uztur abas — SGML un XML — DocBook versijas.

DocBook ir apjomīgs un robusts DTD, un tas var tikt izmantots visdažādāko grāmatu un publikāciju, ne tikai datorliteratūras (kam tas sākotnēji bija paredzēts) rakstīšanā. DocBook lietotāju skaits pasaulē arvien pieaug. Daudzas organizācijas pasaulē izmanto DocBook, rakstot miljoniem dokumentācijas lappušu. [13, 14]

Tā kā DocBook ir XML paveids, tas ir ļoti piemērots strukturētu dokumentu rakstīšanai. Vēl vairāk, tas ir orientēts tieši uz dokumenta satura un loģiskās struktūras izveidi, un minimāli vai nemaz uz konkrētām prezentācijas formām. DocBook formātā saturs ir vēl vairāk nodalīts no formas kā \LaTeX dokumentos. Saturs iegūst formu tikai tad, kad tas ar XSLT²¹ transformāciju palīdzību tiek konvertēts uz citiem formātiem. Jāpiezīmē, ka vienu un to pašu DocBook avota dokumentu var transformēt ar dažādām XSLT transformācijām un iegūt dažādus gala formātus, piemēram, HTML, FO²² un \LaTeX dokumentus.

DocBook jau sen plaši izmanto atklātā pirmkoda pasaulē. Daudzu atklātā pirmkoda produktu dokumentācijas tiek rakstītas un uzturētas DocBook formātā, un tālāk tiek konvertētas dažādos citos formātos. Kā DocBook lietotājus var minēt vairākus linux distributīvus, piemēram, openSUSE²³, kā arī daudzus citus projektus, piemēram, KDE²⁴.

5.2 DocBook pielietojums e-mācībās

XML formātu izmantošana (tai skaitā DocBook) parādās arī e-mācību jomā. Var minēt dažus piemērus:

- ADRIAN sistēmā [17] kā avota dokumenti tiek izmantoti XML faili ar pašu izstrādātu DTD. Sistēma no XML dokumentiem var ģenerēt trīs veida materiālus:

1. Lekciju materiālus (lekciju piezīmes, laboratorijas darbu aprakstus, uzdevumu lapas un c.)
2. Testus un eksāmenus

¹⁹Angl. *Document Type Definition*

²⁰*Organization for the Advancement of Structured Information Standards* — internacionāls, bezpeļņas konsorcijs, kas nodarbojas ar e-biznesa standartu izstrādi, apvienošanu un adaptāciju; <http://www.oasis-open.org/who/>

²¹*Extensible Stylesheet Language Transformation* — XML transformāciju fails

²²*Formatting Objects* — starpposms drukāto materiālu (PDF) izveidei. Ja tiek izmantoti FO, \LaTeX nav nepieciešams, lai noģenerētu PDF dokumentus.

²³<http://www.opensuse.org>

²⁴KDE ir logu pārvaldnieks UNIX operētājsistēmās; <http://www.kde.org>

3. Prezentācijas.

- ❑ Mollojs no Dublinas Pilsētas Universitātes²⁵ Īrijā ir izveidojis objektorientētās programmēšanas kursu, izmantojot DocBook kā avota dokumentu formātu [18]. Izstrādātā sistēma ļauj konvertēt avota dokumentus uz PDF un HTML formātiem. Sistēma dinamiski aizsargā PDF dokumentus, tiem pievienojot studentu personīgo identifikācijas informāciju, un kriptējot tos.

5.3 DocBook dokumentu procesēšanas shēma

Kā jau tika minēts iepriekš, DocBook formāts ļauj gandrīz pilnīgi nodalīt saturu no formas. Tas nozīmē, ka pašā DocBook dokumentā ir minimāla formatēšanas informācija, bet ir ļoti precīza struktūra.

DocBook dokumenti iegūst formu tikai tad, kad tos transformē uz citiem formātiem. Ja jāiegūst PDF, tad DocBook dokuments vispirms jākonvertē uz formatēšanas objektiem FO vai \LaTeX , un pēc tam FO vai \LaTeX dokumenti jāpārveido uz PDF. DocBook dokumentus uz HTML formātu var konvertēt tieši. Šis konversijas ir iespējams automatizēt.

Līdzīgi kā ar citiem XML failiem, DocBook transformācijas tiek veiktas ar speciālu transformācijas failu, XSLT²⁶, palīdzību. XSLT failos tiek aprakstīts, kādā veidā transformēt un formatēt katru DocBook dokumenta elementu. No viena un tā paša avota dokumenta, pielietojot dažādas transformācijas, ir iespējams iegūt dažādus gala dokumentus.

Īsumā DocBook dokumenta izveides ceļš ir šāds:

1. DocBook rediģēšana (parasti ar XML redaktoru). XML redaktoros bieži izmanto iebūvēto XML pārbaudītāju (validātoru), kas pārbauda dokumenta struktūras atbilstību DocBook DTD (skat. 5.3.1. apakšpunktu)
2. Pārbaudītā DocBook dokumenta transformācija uz citiem formātiem, izmantojot XSLT transformācijas. HTML dokumenti tiek ģenerēti tieši, savukārt PDF dokumenti pastarpināti — sākumā teik veidoti formatēšanas objektu (FO) vai \LaTeX faili, un tad tie tiek attiecīgi transformēti par PDF (skat. piemēru 5.4. apakšpunktā)

Gan DocBook DTD, gan arī XSLT transformācijas ir iespējams pielāgot. No DTD ir iespējams izņemt nevajadzīgos tagus, kā arī pievienot jaunus tagus un atribūtus.

5.3.1 DocBook rediģēšana

DocBook ir iespējams rediģēt vairākos veidos, un DocBook rediģēšanai pastāv daudzi redaktori, kas pieejami gan bez maksas, gan arī par maksu. Labam DocBook redaktoram ir jāspēj pārbaudīt dokumenta sintaksi tā rakstīšanas laikā, un tam ir jāprot strādāt ar pielāgoto DocBook DTD. [15]

Īsumā, DocBook redaktorus var iedalīt divās kategorijās:

- A. WYSIWYG²⁷ tipa reaktori, kas atļauj rediģēt DocBook dokumentus vizuāli.

Var nosaukt šādus WYSIWYG tipa redaktorus:

- ❑ OpenOffice²⁸ un AbiWord²⁹. Abas tās ir bezmaksas ofisa aplikācijas, kas prot atvērt un saglabāt arī DocBook dokumentus. Tomēr šo programmu DocBook atbalsts nav pilnīgs;

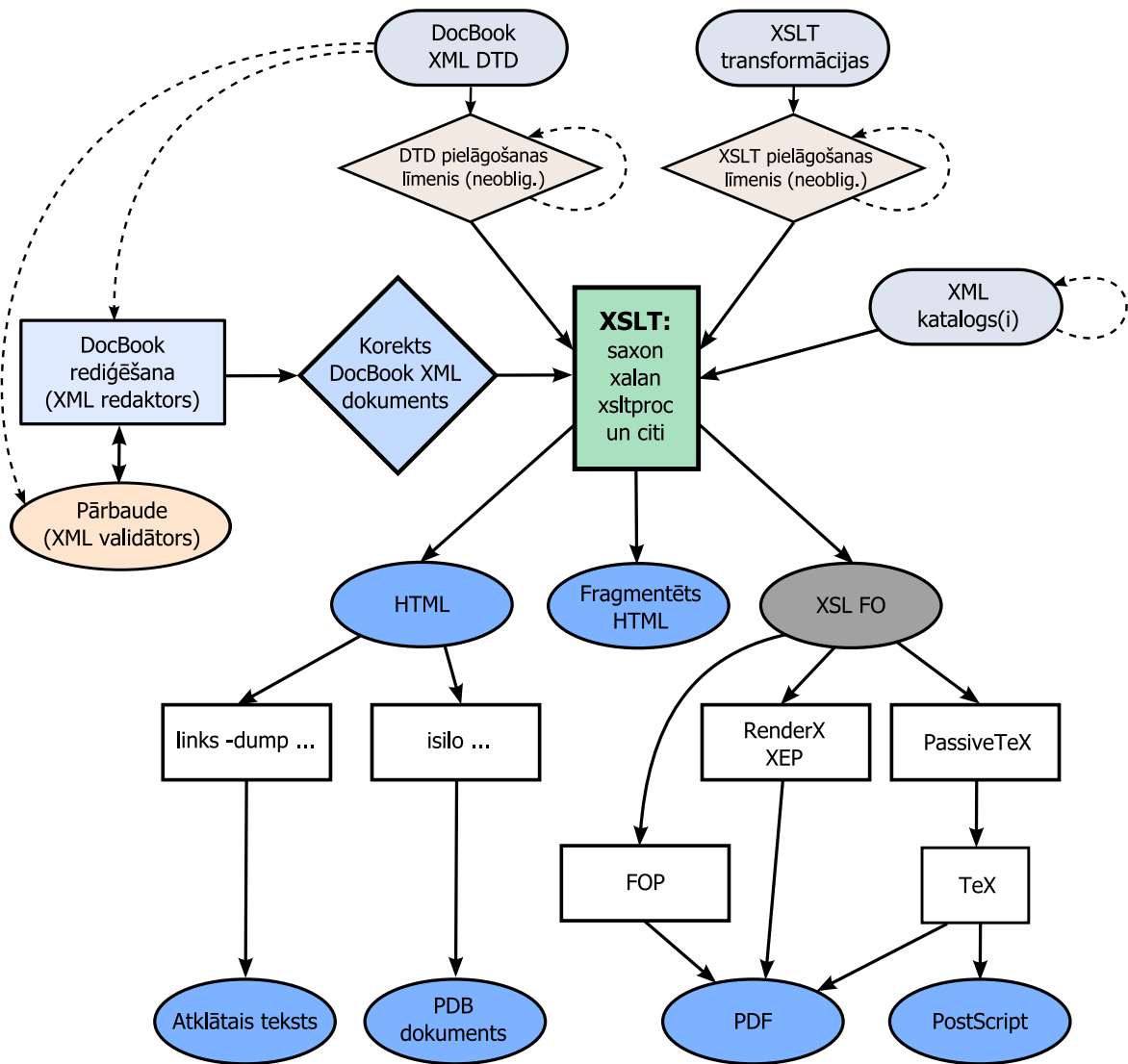
²⁵Dublin City University

²⁶<http://www.w3.org/Style/XSL/>

²⁷*What You See Is What You Get* — ko jūs redzat, to jūs arī iegūstat

²⁸<http://www.openoffice.org/>

²⁹<http://www.abisource.com/>



Att. 3: DocBook procesēšanas shēma [20]

- oXygen³⁰ — jaudīgs maksas XML redaktors, kas ļauj ģenerēt DocBook PDF un HTML;
- Kā tipisku vizuālā redaktora piemēru var minēt XMLmind³¹. Programmas standarta versija nekomerciāliem nolūkiem ir pieejama par brīvu. Programmas ekrānkopija redzama 5. attēlā, 15. lappusē. No redaktora iespējams ģenerēt HTML, RTF, PDF dokumentus
- Syntex Serna³² — jaudīgs WYSIWYG tipa DocBook redaktors

B. XML koda redaktori, kas spēj dinamiski pārbaudīt dokumenta struktūras atbilstību DocBook DTD

Var nosaukt šādus XML redaktorus:

- jEdit³³ — labs atklātā pirmkoda redaktors ar 7 gadu izstrādes vēsturi. Tam

³⁰<http://www.oxygenxml.com/>

³¹<http://www.xmlmind.com/home.html>

³²<http://www.syntext.com/products/serna/index.htm>

³³<http://www.jedit.org/>

ir iespējams pieslēgt XML atbalstu.

- ❑ Exchanger XML Editor³⁴ — vēl viens jaudīgs maksas XML redaktors.

5.3.2 FO vai \LaTeX ?

3. attēlā ir parādīts, ka, lai iegūtu drukāšanas dokumentus, sākumā ir jāveic transformācija uz formatēšanas objektiem FO, un pēc tam, izmantojot kādu no FO renderēšanas dzinējiem, FO ir jātransformē uz PDF. Tomēr pastāv arī alternatīvs varinats — FO vietā izmantot \LaTeX . Katrai metodei ir savas priekšrocības un trūkumi.

Atklātā pirmkoda pasaulē DocBook procesēšanā bieži lieto \LaTeX . DocBook kovertēšanā tiek izmantots DBLaTeX rīks³⁵ vai DB2LaTeX XSLT transformācijas³⁶. Jāpiezīmē, ka raksta autoram neizdevās korekti izmantot minētās programmas ar \LaTeX failiem latviešu valodā, tomēr pastāv varbūtība, ka šīs problēmas var atrisināt.

Apskatīsim sīkāk katru no metodēm.

FO metodes priekšrocības un trūkumi

Priekšrocības

- ✓ Ir pieejamas gatavas XSLT transformācijas, kas transformē DocBook dokumentus uz HTML un FO³⁷;
- ✓ Eksistē vairāki ražošanas kvalitātes FO dzinēji: [16]
 - * Antenna House XSL Formatter³⁸ — šķiet, ka viens no jaudīgākajiem komerciālajiem FO dzinējiem. Spēj renderēt arī MathML formulas;
 - * RenderX XEP³⁹ — vēl viens ražošanas kvalitātes FO dzinējs. Ir pieejama arī bezmaksas versija privātai lietošanai.
- ✓ Eksistē vairāki brīvi pieejami FO dzinēji, daži no tiem tiek attīstīti:
 - * Apache FOP (Formatting Objects Processor)⁴⁰ — iespējams nopietnākais no atklātā pirmkoda FO procesoriem, kas tiek attīstīts;
 - * xmlroff XSL Formatter⁴¹ — nav līdz galam pabeigts, un izskatās, ka vairs netiek attīstīts.
- ✓ Komerciālie FO dzinēji atbalsta Unicode fontus, MathML, SVG, nodrošina ražošanas kvalitāti

Trūkumi

- ✗ FO tehnoloģija, salīdzinot ar \LaTeX , ir jauna, un tās atklātā pirmkoda programmatūras atbalsts vēl nav pilnīgs;
- ✗ Komerciālie FO procesori nodrošina ražošanas kvalitāti, bet ir ļoti dārgi — piemēram, Antenna House XSL Formatter servera versija maksā vairākus tūkstošus dolāru.

³⁴<http://www.exchangerxml.com/index.html>

³⁵<http://dblatex.sourceforge.net/>

³⁶<http://sourceforge.net/projects/db2latex>

³⁷<http://docbook.sourceforge.net/projects/xsl/>

³⁸<http://www.antennahouse.com/>

³⁹<http://www.renderx.com/>

⁴⁰<http://xmlgraphics.apache.org/fop/>

⁴¹<http://xmlroff.sourceforge.net/>

LaTeX metodes priekšrocības un trūkumi

Priekšrocības

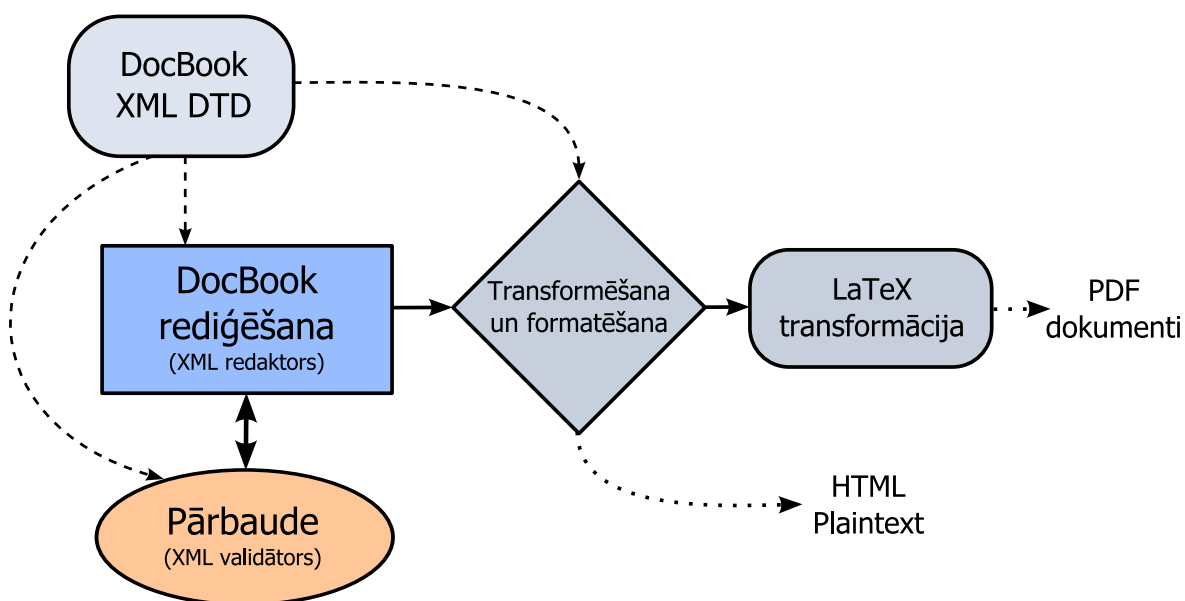
- ✓ LaTeX pastāv jau vairāk kā 20 gadus, un ir ļoti nobriedusi un noslīpēta sistēma
- ✓ DocBook PDF veidošanā ir pieejamas visas LaTeX priekšrocības, kas aprakstītas 4.5. apakšpunktā
- ✓ DocBook kombinācijā ar pašu rakstītām vai pielāgotām LaTeX transformācijām sniedz ļoti elastīgu PDF ģenerēšanas sistēmu (skat. 5.3.3. apakšpunktu)

Trūkumi

- ✗ Tie paši trūkumi, kas aprakstīti 4.5. apakšpunktā;
- ✗ Ir grūtības ģenerēt LaTeX tabulas, jo tās ir mazāk elastīgas, nekā DocBook. DocBook standarts ļauj veidot ļoti sarežģītas tabulas.

Grāmata “A beginner’s introduction to typesetting with LaTeX” [1] tiek rakstīta un uzturēta DocBook formātā, un PDF formāts ģenerēts, izmantojot LaTeX.

Arī šis dokuments ir rakstīts DocBook formātā ar dažiem pielāgojumiem un autora rakstītām XSLT transformācijām uz LaTeX formātu. Raksta autors izmanto DocBook procesēšanas shēmu, kas parādīta 4. attēlā.



Att. 4: Vienkāršota DocBook procesēšanas shēma, izmantojot LaTeX

5.3.3 PDF un HTML dokumentu formatēšana

Veidojot HTML un PDF dokumentus, ir iespējams un vēlams tos formatēt divos līmeņos:

1. līmenis ir formatēšanas pievienošana, veicot XSLT transformācijas uz HTML un LaTeX. Šī līmeņa ietvaros ir jātransformē dokumenta struktūra, kā arī jāpievieno obligāti nepieciešamais formatējums. Šeit būtu jāizvairās pievienot smalkāku formatējumu, piemēram, norādīt fontu;

- līmenis ir \LaTeX un HTML stilu izmantošana. \LaTeX un HTML stilu failos tiek līdz galam norādīts dokumenta noformējums, piemēram, pielāgota sarakstu parādīšana, norādīti fonti, korigēti citi dokumentu elementi. \LaTeX dokumentos, piemēram, var izmantot komandu `\usepackage{mans_stils}`, kur `mans_stils` atbilst stilu failam `mans_stils.sty`.

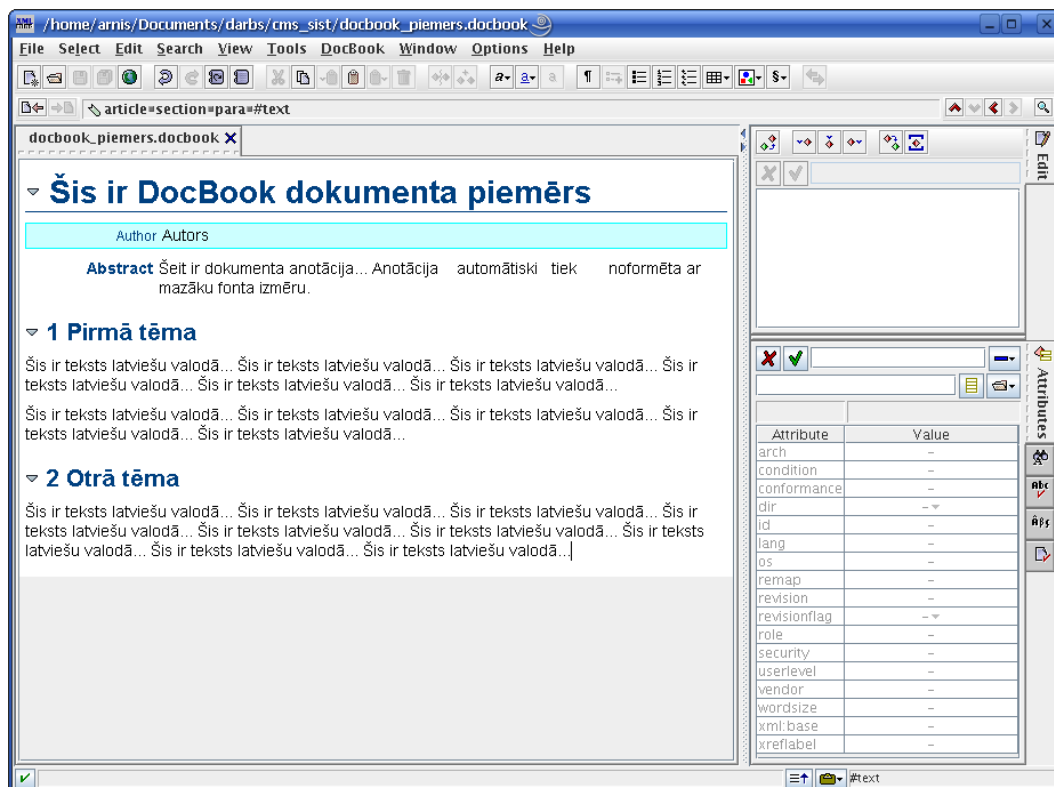
Abu šo formatējuma līmeņu vienlaicīga izmantošana sniedz ārkārtīgu elastību HTML un PDF dokumentu izveidē un noformēšanā. Tādā veidā kontrolēt PDF dokumentu formatējumu ir vēl ērtāk, nekā rakstot \LaTeX dokumentus tieši. Tas ļauj ļoti efektīvi veidot un attīstīt dokumentu korporatīvo stilu, jo, mainot un pilnveidojot dokumentu formatējumu, DocBook avota dokumenti paliek neskarti, un saturs un forma var tikt pilnveidoti paralēli un neatkarīgi viens no otra.

5.4 DocBook piemērs

Lai varētu uzrakstīt un noprosesēt DocBook dokumentu, izmantojot \LaTeX , ir nepieciešamas trīs programmas:

- DocBook redaktors (skat. 5.3.1.)
- XSLT procesors, piemēram, Saxon⁴² vai xsltproc⁴³
- Strādājoša \LaTeX tipogrāfiskā sistēma

Uzrakstīsim DocBook dokumentu, kas pēc struktūras atbilst 5. attēlā sniegtajam piemēram, izmantojot XMLmind redaktoru.



Att. 5: XMLmind WYSIWYG DocBook redaktora ekrānkopija

⁴²<http://saxon.sourceforge.net/>

⁴³<http://xmlsoft.org/XSLT/>

Saglabājot dokumentu, tiek izveidots DocBook fails (skat. **A. pielikumu**).

Tagad fails ir jāpārveido \LaTeX formātā, un tam ir nepieciešams XSLT skripts.

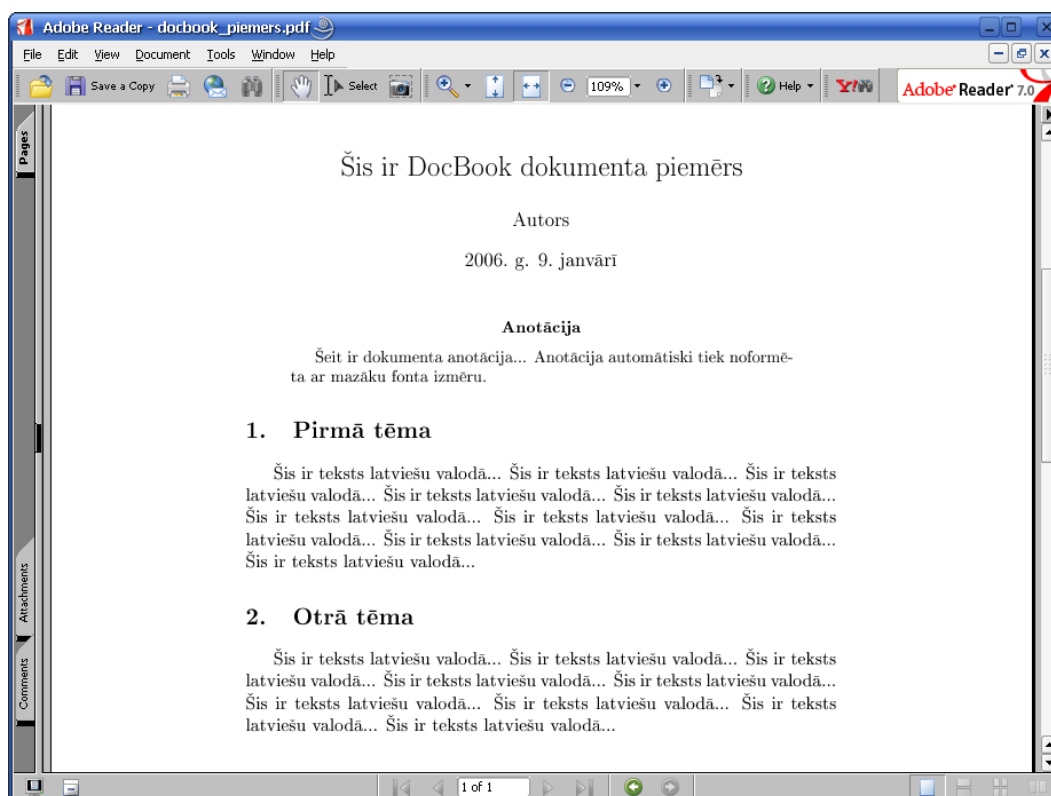
XSLT skriptus rakstīt nav pārāk sarežģīti, tomēr tas ir jāapgūst. XSLT failā secīgi jāapraksta darbības, kas jāveic ar katru XML tagu. Šajā piemērā izmantotais XSLT fails redzams **B. pielikumā**.

Transformēsim dokumentu ar Saxon XSLT procesoru. Izpildām komandu:

```
1 > saxon -o docbook_piemers.tex docbook_piemers.docbook docbook_piemers.xsl
```

Pēc komandas izpildes tiek izveidots fails `docbook_piemers.tex` (skat. **C. pielikumu**), kas tālāk jānokompilē ar komandu:

```
1 > pdflatex docbook_piemers.tex
```



Att. 6: Izveidotais DocBook piemēra PDF dokuments

Piemērā izveidotā dokumenta ekrānkopija redzama **6. attēlā**.

5.5 Matemātika DocBook un LaTeX dokumentos

DocBook dokumentos var rakstīt matemātikas formulas \LaTeX formātā, un, transformējot dokumentu uz PDF, \LaTeX tās interpretē kā parasti.

Tomēr šī paņēmiena trūkums ir tāds, ka, transformējot DocBook uz citiem formātiem, formulas ir arī jātransformē, bet ar XSLT transformācijām \LaTeX formulas transformēt nav iespējams (jo tas nav XML). Ir jāmeklē citas iespējas, kā transformēt \LaTeX formulas. Jāpiezīmē, ka pastāv rīki, kas prot konvertēt \LaTeX formulas uz MathML.

Tomēr daudz labāks variants ir materiālu avota dokumentos (DocBook) likt formulas MathML formātā.

MathML ir XML valoda, un tā ir W3C rekomendētā valoda matemātikas formulu rakstīšanai. MathML ir paredzēta, lai padarītu iespējamu matemātikas apmaiņu starp programmām, un lai sniegtu pamatus matemātikas ievietošanai tīmeklī.

MathML, salīdzinot ar \LaTeX formulu pierakstu, ir jauns standarts — MathML 2.0 rekomendācijas iznāca 2001. gada sākumā. Šobrīd MathML atbalsta daudzi rīki. [21, 22]

Formulas MathML formātā ar roku praktiski nav rediģējamas (vai arī to darīt ir ļoti neērti un nepārskatāmi), tādēļ rediģēšanai ir jāizmanto rīki. Ne visi no pieejamie rīkiem ir labi, bet daži no tiem ir ļoti noderīgi. Kā ļoti noderīgu MathML formulu rediģēšanas un publicēšanas rīku var minēt WebEQ [3].

MathML formulu priekšrocība ir tāda, ka tās var transformēt kopā ar DocBook dokumentiem. Ir pieejamas brīnišķīgas MathML uz \LaTeX XSLT transformācijas [23]. Jāpiebilst, ka, ja PDF dokumentu veidošanā izmanto formatēšanas objektus (FO), ir FO procesori, kas spēj procesēt MathML tieši — piemēram, Antenna House XSL Formatter.

Ja tiek veidoti HTML dokumenti, formulas var netransformēt, jo pārlūkprogrammas spēj parādīt MathML formulas (uzreiz vai ar spraudņu palīdzību). Pastāv arī iespēja reprezentēt formulas kā attēlus, tomēr MathML izmantošana ir pats labākais veids, kā likt formulas tīmeklī, tam ir daudzas priekšrocības.

5.5.1 MathML izmantošanas piemērs

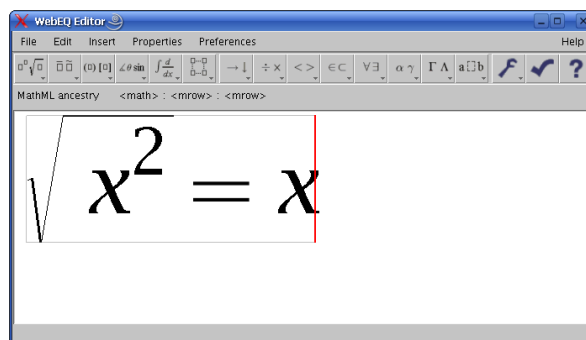
Izmantojot WebEQ redaktoru, iespējams uzrakstīt matemātikas formulu (7. att.), to nokopēt MathML formātā un ielīmēt DocBook dokumentā. Ja nepieciešams, ir iespējams no DocBook dokumenta nokopēt MathML formulu, ielīmēt to atpakaļ WebEQ redaktorā un vizuāli to rediģēt.

Nokopējam formulu no WebEQ redaktora un ielīmējam to DocBook dokumentā:

```
1 <mml:math>
2   <mml:mrow>
3     <mml:mrow>
4       <mml:msqrt>
5         <mml:mrow>
6           <mml:msup>
7             <mml:mrow>
8               <mml:mi>x</mml:mi>
9             </mml:mrow>
10            <mml:mrow>
11              <mml:mn>2</mml:mn>
12            </mml:mrow>
13          </mml:msup>
14        </mml:mrow>
15      </mml:msqrt>
16    </mml:mrow>
17    <mml:mo>=</mml:mo>
18    <mml:mrow>
19      <mml:mi>x</mml:mi>
20    </mml:mrow>
21  </mml:mrow>
22 </mml:math>
```

Transformējot DocBook dokumentu ar DocBook un MathML XSLT transformācijām, formula tiek konvertēta uz \LaTeX formātu:

```
1 \sqrt{{x}^{2}}=x
```



Att. 7: Formula WebEQ redaktorā

Šāds ir gala rezultāts, kad \LaTeX dokuments tiek noprocēsēts: $\sqrt{x^2} = x$
 Papildinot DocBook DTD ar papildus elementu `<latexmath>`, \LaTeX formulas DocBook tekstā var ierakstīt tieši:

1 `<latexmath>\sqrt{{x}^{2}}=x</latexmath>`

Rezultāts ir tieši tāds pats: $\sqrt{x^2} = x$.
 Kaut arī šajā piemērā tika parādīta vienkārša formula, līdzīgā veidā ir iespējams rediģēt un ievietot DocBook dokumentos arī ļoti komplicētas formulas.

5.6 DocBook priekšrocības un trūkumi

Apkoposim DocBook kā e-mācību materiālu avota failu formāta priekšrocības un trūkumus.

Priekšrocības:

- ✓ DocBook ļauj pilnīgi nodalīt saturu no formas (vēl labāk kā \LaTeX dokumentos), tādā veidā sniedzot iespēju standartizēt materiālu formatējumu.
- ✓ DocBook ir pielāgojams formāts — ir iespējams definēt savus struktūras elementus, mainīt tagu atribūtus, vai izņemt ārā nevajadzīgos tagus
- ✓ DocBook uztur UTF-8 kodējumu
- ✓ No DocBook ir iespējams ģenerēt dokumentus daudzos formātos — HTML, PDF, PDB, PostScript
- ✓ Ir daudz paņēmieni un rīku DocBook dokumentu rediģēšanai un transformēšanai — gan par maksu, gan arī brīvi pieejami
- ✓ DocBook popularitāte pieaug, un ar laiku var parādīties vēl vairāk rīku darbam ar DocBook

Trūkumi:

- ✗ Brīvi pieejamu DocBook WYSIWYG redaktoru ir samērā maz
- ✗ Tā kā konvertāciju ir tikt daudz, nav viena pareizā transformāciju mehānisma. Tomēr ir iespējams izveidot vienotu sistēmu uzņēmumā vai iestādē
- ✗ Izveidot strādājošu DocBook publicēšanas sistēmu var būt darbietilpīgi un laikietilpīgi [18]

- ✗ DocBook var būt par sarežģītu, jo tajā ir ļoti daudz struktūras elementu. Tomēr, veidojot konkrētas DocBook implementācijas, ir iespējams izņemt nelietotos tagus un atribūtus
- ✗ Sīkāk DocBook dokumentu procesēšanas priekšrocības un trūkumi ir aprakstīti 5.3.2. apakšnodaļā

DocBook formāts ļoti labi atbilst 2. nodaļā aprakstītajām e-kursa materiālu avota failu formāta vispārīgajām prasībām.

6 Nobeigums

Šajā pārskatā ir sīki apskatīti divi formāti, kuri varētu kalpot par e-mācību materiālu avota formātu — \LaTeX un DocBook, kā arī pieminēta datubāzu izmantošanas iespējamība.

\LaTeX var kalpot par avota formātu, tomēr ar to būtu saistītas vairākas grūtības, piemēram, \LaTeX dokumentu konvertēšana uz citiem, ne PDF, formātiem.

Turpretīm DocBook formāts paver ļoti plašas iespējas dokumentu transformēšanā. Ir iespējams DocBook lietot kopā ar \LaTeX , izmantojot visas labās \LaTeX īpašības PDF dokumentu izveidē, kā arī saglabājot iespēju ērti ģenerēt HTML dokumentus. Bez \LaTeX , DocBook procesēšanai pastāv arī citas, komerciālas tehnoloģijas.

Viennozīmīgi, DocBook formāts ir elastīgāks par \LaTeX formātu, un kopumā piedāvā vairāk iespēju. Tas ir formāts, kurā var un vajag izstrādāt e-kursu materiālu oriģinālus, tādā veidā atvieglojot materiālu tālāku papildināšanu un uzturēšanu.

Literatūra

- [1] Flynn P. **Formatting information. A beginner's introduction to typesetting with \LaTeX .** <http://www.ctan.org/tex-archive/info/beginlatex/>, pēdējais apmeklējums 16.12.2005..
- [2] Oetiker T., Partl H., Hyna I., Schlegl E. **A (Not So) Short Introduction to \LaTeX 2e.** <http://www.ctan.org/info?id=lshort-english>.
- [3] WebEQ. **Interactive Math on the Web. Design Science,** <http://www.dessci.com/en/products/webeq/>, pēdējais apmeklējums 16.12.2005..
- [4] UTF-8. **Wikipedia,** <http://en.wikipedia.org/wiki/UTF-8>.
- [5] \LaTeX . **Wikipedia,** <http://en.wikipedia.org/wiki/LaTeX>.
- [6] \TeX . **Wikipedia,** <http://en.wikipedia.org/wiki/TeX>.
- [7] XeTeX. **Wikipedia,** <http://en.wikipedia.org/wiki/XeTeX>.
- [8] **The XeTeX typesetting system. NRSI: Computers Writing Systems,** http://scripts.sil.org/cms/scripts/page.php?site_id=nrsi&id=xetex.
- [9] **Just What Is TeX?. \TeX users group,** <http://www.tug.org/whatis.html>.
- [10] **\LaTeX 2HTML,** <http://saftsack.fs.uni-bayreuth.de/~latex2ht/>.
- [11] **\LaTeX and \TeX for Hypertext. \TeX 4ht,** <http://www.cse.ohio-state.edu/~gurari/TeX4ht/>.
- [12] **Driķis I. \LaTeX latviskošana.** <http://home.lanet.lv/~drikis/TeX/index.html>.

- [13] **What Is DocBook?. OASIS,**
<http://www.oasis-open.org/committees/docbook/intro.shtml>.
- [14] **DocBook. DocBook Wiki,** <http://wiki.docbook.org/topic/DocBook>.
- [15] **DocBookAuthoringTools. DocBook Wiki,**
<http://wiki.docbook.org/topic/DocBookAuthoringTools>.
- [16] **DocBook Publishing Tools. DocBook Wiki,**
<http://wiki.docbook.org/topic/DocBookPublishingTools>.
- [17] **Librelotto G.R., Ramalho J.C., Henriques P.R. ADRIAN. E-learning Content Production (creating online exams). Universidade do Minho, Braga, Portugal,**
<http://www.di.uminho.pt/~jcr/XML/publicacoes/artigos/2004/RLH04.pdf>,
2004.
- [18] **Molloy D. Single-Source Interactive and Printed Content Publishing Using the DocBok XML Standard. Dublin City University,** http://www.eeng.dcu.ie/~molloyd/research/DMolloy_PublishingUsingDocBook.pdf.
- [19] **Barahona J.G., Dimitrova V., Chaparro D., Tebb C., Mazza R. Producing educational resources int the “libre”way. The Edukalibre project. Edukalibre project,**
http://edukalibre.org/documentation/edukalibre_system.pdf.
- [20] **Walsh N. DocBook Publishing Model.** <http://nwalsh.com/docbook/procdiagram/>.
- [21] **W3C Math Home. W3C,** <http://www.w3.org/Math/>.
- [22] **The W3C MathML software list. W3C,** <http://www.w3.org/Math/Software/>.
- [23] **Yaroshevich V. XSLT MathML Library.** <http://xsltml.sourceforge.net/>.

A DocBook fails

docbook_piemers.docbook

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE article PUBLIC "-//OASIS//DTD DocBook XML V4.4//EN"
3 "http://www.oasis-open.org/docbook/xml/4.4/docbookx.dtd">
4 <article>
5   <title>Šis ir DocBook dokumenta piemērs</title>
6
7   <articleinfo>
8     <author>
9       <surname>Autors</surname>
10    </author>
11  </articleinfo>
12
13  <abstract>
14    <para>Šeit ir dokumenta anotācija... Anotācija automātiski tiek noformēta
15    ar mazāku fonta izmēru.</para>
16  </abstract>
17
18  <section>
19    <title>Pirmā tēma</title>
20
21    <para>Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā...
22    Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir
23    teksts latviešu valodā... Šis ir teksts latviešu valodā...</para>
24
25    <para>Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā...
26    Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir
27    teksts latviešu valodā...</para>
28  </section>
29
30  <section>
31    <title>Otrā tēma</title>
32
33    <para>Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā...
34    Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir
35    teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts
36    latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts latviešu
37    valodā...</para>
38  </section>
39 </article>
```

B XSLT transformāciju fails

docbook_piemers.xsl

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <xsl:stylesheet
3   xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
4
5 <xsl:output method="text" encoding="utf-8"/>
6
7 <!-- ===== -->
8 <!-- Dokumenta sākums -->
9 <xsl:template match="/">
10   <xsl:apply-templates/>
11 </xsl:template>
12
13 <!-- Raksts -->
14 <xsl:template match="article">
15   <xsl:text>\documentclass[a4paper,11pt]{article}&#x000A;</xsl:text>
16   <xsl:text>\usepackage[utf8]{inputenc}&#x000A;</xsl:text>
17   <xsl:text>\usepackage[LV]{fontenc}&#x000A;</xsl:text>
18   <xsl:text>\usepackage[latvian]{babel}&#x000A;&#x000A;</xsl:text>
19
20   <xsl:text>\begin{document}&#x000A;</xsl:text>
21   <xsl:apply-templates/>
22   <xsl:text>\end{document}&#x000A;</xsl:text>
23 </xsl:template>
24
25 <!-- ===== -->
26 <!-- Virsraksts -->
27 <xsl:template match="article/title">
28   <xsl:text>\title{</xsl:text>
29   <xsl:apply-templates/>
30   <xsl:text>}&#x000A;</xsl:text>
31 </xsl:template>
32
33 <xsl:template match="articleinfo">
34   <xsl:apply-templates/>
35   <xsl:text>\maketitle&#x000A;</xsl:text>
36 </xsl:template>
37
38 <xsl:template match="author">
39   <xsl:text>\author{</xsl:text>
40   <xsl:value-of select="firstname"/>
41   <xsl:text> </xsl:text>
42   <xsl:value-of select="surname"/>
43   <xsl:text>}&#x000A;</xsl:text>
44 </xsl:template>
45
46 <!-- ===== -->
47 <!-- Anotācija -->
48 <xsl:template match="abstract">
49   <xsl:text>&#x000A;\begin{abstract}</xsl:text>
50   <xsl:apply-templates/>
51   <xsl:text>&#x000A;\end{abstract}&#x000A;</xsl:text>
52 </xsl:template>
53
54 <!-- ===== -->
55 <!-- Nodaļas un paragrāfi -->
56 <xsl:template match="section">
57   <xsl:text>&#x000A;</xsl:text>
58   <xsl:apply-templates/>
59   <xsl:text>&#x000A;</xsl:text>
60 </xsl:template>
61
62 <xsl:template match="section/title">
63   <xsl:text>\section{</xsl:text>
64   <xsl:apply-templates/>
65   <xsl:text>}&#x000A;</xsl:text>
66 </xsl:template>
67
68 <xsl:template match="para">
69   <xsl:text>&#x000A;</xsl:text>
70   <xsl:apply-templates/>
71 </xsl:template>
72
73 </xsl:stylesheet>
```

C Generētais \LaTeX fails

docbook_piemers.tex

```
1 \documentclass[a4paper,11pt]{article}
2 \usepackage[utf8]{inputenc}
3 \usepackage[LV]{fontenc}
4 \usepackage[latvian]{babel}
5
6 \begin{document}
7 \title{Šis ir DocBook dokumenta piemērs}
8 \author{Autors}
9 \maketitle
10
11 \begin{abstract}
12 Šeit ir dokumenta anotācija... Anotācija automātiski tiek noformēta
13 ar mazāku fonta izmēru.
14 \end{abstract}
15
16 \section{Pirmā tēma}
17
18 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā...
19 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir
20 teksts latviešu valodā... Šis ir teksts latviešu valodā...
21 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā...
22 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir
23 teksts latviešu valodā...
24
25 \section{Otrā tēma}
26
27 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā...
28 Šis ir teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir
29 teksts latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts
30 latviešu valodā... Šis ir teksts latviešu valodā... Šis ir teksts latviešu
31 valodā...
32 \end{document}
```
