

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

AUTOMATIZOVANÉ GENEROVÁNÍ SBORNÍKU KONFERENCE
POMOCÍ XML

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

BRANISLAV REPKA

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

AUTOMATIZOVANÉ GENEROVÁNÍ SBORNÍKU KONFERENCE POMOCÍ XML

XML-BASED AUTOMATED GENERATION OF CONFERENCE PROCEEDINGS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

BRANISLAV REPKA

VEDOUCÍ PRÁCE

SUPERVISOR

Mgr. PAVEL RAJMIC, Ph.D.

BRNO 2015



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Branislav Repka

ID: 155227

Ročník: 3

Akademický rok: 2014/2015

NÁZEV TÉMATU:

Automatizované generování sborníku konference pomocí XML

POKYNY PRO VYPRACOVÁNÍ:

Konferenci většího rozsahu je nutné doprovázet dodatečnými materiály. Zjistěte jaké existují systémy pro zajištění těchto cílů a porovnejte jejich vlastnosti. Vyberte nejlepšího zástupce a proveďte jeho použití, modifikaci a doplnění tak, aby bylo možné tento systém použít pro pořádání konference DAFx v roce 2016 v Brně. Vytvořte XSL šablonu tak, aby se automatizovaně, užitím HTML, pdflatexu či obojího, vytvořil sborník konference a vizitky k tisku.

DOPORUČENÁ LITERATURA:

[1] Grusová L. XML pro úplné začátečníky. Computer press, 2002.

[2] Rybička, J. LaTeX pro začátečníky. Třetí vydání, Konvoj, 2003.

[3] Konferenční stránka <http://www.dafx.de/>

Termín zadání: 9.2.2015

Termín odevzdání: 2.6.2015

Vedoucí práce: Mgr. Pavel Rajmic, Ph.D.

Konzultanti bakalářské práce:

doc. Ing. Jiří Mišurec, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Tento dokument sa zaoberá vytváraním materiálov pomocou XML a XSL pre konferencie. Konkrétne, automatické generovanie zborníka konferencie vo formáte PDF a HTML stránky a vizitiek pre usporiadanie konferencie. V tejto práci sú popísané technológie XML a ich využitie. Tak isto práca zahrňuje návrh a vytvorenie vizitiek a zborníka konferencie pomocou XML technológií.

KĽÚČOVÉ SLOVÁ

XML technológie, XSL a XSLT transformácie, DTD definície, konferenčné systémy

ABSTRACT

This document deals with creating materials using XML and XSL for the conference. Specifically, Automated generation Conference proceedings in format PDF and HTML page and business cards to organize a conference. This paper describes the XML technology and their applications. Also work includes the design and creation of business cards and conference proceedings using XML technologies.

KEYWORDS

XML technologies, XSL and XSLT transformation, DTD definition, conference management systems

REPKA, Branislav *Automatizované generovanie zborníku konferencie pomocou XML*: bakalárska práca. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačných technológií, Ústav telekomunikací, 2015. 48 s. Vedúci práce bol Ing. Pavel Rajmic, Ph.D.

PREHLÁSENIE

Prehlasujem, že som svoju bakalársku prácu na tému „Automatizované generovanie znorníku konferencie pomocou XML“ vypracoval(a) samostatne pod vedením vedúceho bakalárskej práce, využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor(ka) uvedenej bakalárskej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto bakalárskej práce som neporušil(a) autorské práva tretích osôb, najmä som nezasiahol(-la) nedovoleným spôsobom do cudzích autorských práv osobnostných a/nebo majetkových a som si plne vedomý(-á) následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., o právu autorskom, o právach súvisajúcich s právom autorským a o zmeně niektorých zákonov (autorský zákon), vo znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákoníka č. 40/2009 Sb.

Brno

.....

podpis autora(-ky)

POĎAKOVANIE

Rád by som poďakoval vedúcemu bakalárskej práce pánovi Ing. Pavlovi Rajmicovi, Ph.D. za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci.

Brno

.....

podpis autora(-ky)



Faculty of Electrical Engineering
and Communication
Brno University of Technology
Purkynova 118, CZ-61200 Brno
Czech Republic
<http://www.six.feec.vutbr.cz>

POĎAKOVANIE

Výzkum popsaný v tejto bakalárskej práci bol realizovaný v laboratóriách podporených projektom SIX; registračné číslo CZ.1.05/2.1.00/03.0072, operačný program Výzkum a vývoj pro inovace.

Brno

.....

podpis autora(-ky)



EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI



OBSAH

Úvod	10
1 Konferenčné systémy	11
1.1 Konferencie	11
1.2 Vedecké konferencie	12
1.2.1 Typy vedeckých konferencií	12
1.2.2 Príprava vedeckej konferencie	13
1.3 Konferenčné systémy	14
1.3.1 Open-source	14
1.3.2 Peer review	14
1.4 Výber konferenčného systému	15
1.4.1 Prehľad konferenčných systémov	15
1.4.2 Výber konferenčného systému	16
1.5 Konferencia DAFx	18
2 Využívané technológie	20
2.1 Jazyk PHP	20
2.2 Jazyk XML	20
2.3 Definícia typu dokumentu DTD	21
2.4 Jazyk CSS	22
2.5 Jazyk XSL	23
2.6 MySQL	24
2.7 L ^A T _E X	24
3 Popisované technológie použité v praxi	25
3.1 Použitie PHP	25
3.2 Použitie jazyka XML v našej práci	25
3.3 Použitie DTD	27
3.4 Použitie jazyka XSL	28
3.4.1 XSL pre automatizované vytváranie účastníckych kariet	29
3.4.2 Automatizované vytváranie HTML stránky príspevkov	30
3.4.3 XSL pre vytvorenie automatizovaného PDF zborníka	32
3.5 Výsledky práce pomocou procesoru Saxon	35
4 Výsledky študentskej práce	37
4.1 Účastnícke karty	37
4.2 HTML zborník konferencie	37
4.3 PDF zborník konferencie	38

5 Záver	39
Literatúra	40
Zoznam symbolov, veličín a skratiek	43
Zoznam príloh	45
A Ukážka vytvoreného PDF zborníku	46
B Obsah priloženého CD	47

ZOZNAM OBRÁZKOV

1.1	DAFx	18
3.1	DTD v hlavičke dokumentu XML	28
3.2	Hlavička XSL dokumentu	29
3.3	Hlavička dokumentu XML	31
3.4	Príkazy pre automatické vkladanie príspevkov a obsahu	35
3.5	Vytvorenie HTML stránky procesorom Saxon	36
3.6	Vytvorenie PDF súboru pomocou procesora Saxon	36
4.1	Vytvorená účastnícka karta	37
4.2	Časť HTML zborníka konferencie	38
A.1	Obal zborníku konferencie	46
A.2	Časť vytvoreného zborníka	46

ÚVOD

Zmyslom práce je podpora a činnosť spojená práve s prípravou konferencie a vydaním zborníka a účastníckych kariet, automatizovanie vo veľkej miere pomocou jazykov XML(eXtensible Markup Language), PHP(Hypertext Preprocessor). Šablóny budú vytvorené jazykom XSL(eXtensible Stylesheet Language). Vysádzanie výsledných textov podľa typografických zásad zabezpečíme pomocou \LaTeX u.

Prvá časť semestrálnej práce je venovaná konferenčným systémom(ďalej len CMS z anglického „Conference Managment Systems“), ich všeobecnému použitiu, ukážeme si, čo nám CMS poskytujú. Na základe získaných poznatkov porovnáme známe CMS a vyberieme systém, ktorý našim požiadavkám najviac vyhovuje. Popíšeme konferenciu, ktorú usporadúvame.

V ďalšej časti sa oboznámime s jazykom PHP, ktorý ale nieje východiskovým pre našu prácu, takže sa mu nebudeme venovať do hĺbky. Ďalej sa práca venuje jazyku XML, ktorého hlavnou ideou je oddelenie obsahu od vzhľadu súboru. Jazyk XML nám poskytuje veľa možností pre prácu s ním. Preberieme si jeho hlavné výhody, ktoré využijeme v našej práci a tak isto aj nevýhody tohoto jazyku. Tak isto veľmi podstatná bude pre nás nasledujúca časť v ktorej sa budeme venovať vyššie spomínanej výhode, oddeleniu obsahu od vzhľadu, no tento raz grafickej časti a to pomocou jazyku XSL a XSLT. Tieto jazyky sú pre našu prácu veľmi dôležité, pretože nám popisujú ako majú byť nami zadané súbory vo formáte XML zobrazené, teda popisujú nám celkový grafický vzhľad. Budeme sa venovať tvorbe šablóny XSL pre zborník, účastnícke karty a HTML stránku.

Avšak celkovú úpravu finálnych textových súborov nám v konečnej verzii pomôže zabezpečiť \LaTeX , ktorý si popíšeme v ďalšej časti práce. V posledných častiach práce popíšeme využitie spomínaných jazykov v našej práci a tiež celkový postup pri riešení automatizovaného usporadúvania konferencie, nami získane poznatky, výhody a nevýhody nami zvoleného postupu tvorenia konferencie.

1 KONFERENČNÉ SYSTÉMY

1.1 Konferencie

Za konferenčný systém považujeme software ktorý ľuďom napomáha k organizácii konferencií. Konferencia je obvykle kratšie diskusné stretnutie ľudí, kompetentných osôb, k určitej téme. Môžu byť jednorázové, alebo periodické. Niekedy môže byť konferencia aj trvalá inštitúcia. Poznáme nasledujúce typy konferencií: [12]

- **Tlačová konferencia** je stretnutie jednotlivcov, alebo predstaviteľov firiem, či iných organizácií spolu so zástupcami médií. Usporadúva sa buď pravidelne, napríklad pravidelné tlačové konferencie predstaviteľov vlády, alebo vo chvíli, keď je potrebné prezradiť dôležité informácie verejnosti. Po konferencii väčšinou nasleduje diskusia, ktorá umožňuje novinárom klásť otázky.[22]
- **Elektronická konferencia** je spoločenské stretnutie ľudí z rôznych odborov vo svete internetu. Ľudia spolu komunikujú a zdieľajú informácie prostredníctvom internetu. Touto cestou si navzájom predávajú rady, či diskutujú o problematike. Do tohoto typu konferencií môžeme zaradiť aj video konferencie. Je to vzdialená interaktívna komunikácia medzi dvoma, alebo viacerými účastníkmi. Dochádza medzi nimi k prenosu zvukovej a obrazovej informácie cez internet.[5]
- **E-mailová konferencia** je spôsob diskutovania o určitej problematike pomocou elektronickej pošty cez internet. E-mailová konferencia je vytvorená tak, že e-mailové správy zaslané z e-mailovej adresy člena konferencie na určitú e-mailovú adresu sú automaticky rozosielené všetkým členom zaregistrovaným v príslušnej diskusnej skupine.[6]
- **Biskupská konferencia** je väčšinou zbor katolíckych biskupov daného štátu, alebo územia. Územne býva biskupská konferencia vymedzená obvykle pre štát. Samozrejme sú aj biskupské konferencie vymedzené pre oblasti zahrňujúce väčší počet štátov, z ktorých aspoň niektoré majú málo katolíkov, či obyvateľov. Preto by sa nevyplatilo vytvoriť pre nich samostatnú biskupskú konferenciu.[1]
- **Vedecká konferencia** je obvykle krátkodobé stretnutie odborníkov v určitej oblasti. V našej práci sa budeme venovať práve tomuto typu konferencií. Preto je jej podrobný popis vysvetlený nižšie.

1.2 Vedecké konferencie

Vedecká konferencia je stretnutie odborníkov a vedcov za účelom prezentovania a diskusie ich práce. Je dôležitým miestom pre výmenu informácií a poznatkov medzi vedcami. Často bývajú organizované univerzitami a konané na akademickej pôde. Dôležitou výhodou prezentácie dát na konferencii v porovnaní s ich uverejnením vo vedeckom časopise je, že spätná väzba a následná diskusia prebieha okamžite. Vedecké konferencie vedú väčšinou k vydaniu zborníku, kde sa nachádza zhrnutie výskumných projektov, ktoré boli predložené.

1.2.1 Typy vedeckých konferencií

Vedecké konferencie môžu byť otvorené. Vtedy jednotliví záujemcovia o prezentovanie svojich výsledkov zasielajú organizátorom svoje odborné príspevky. Tieto príspevky sú ďalej hodnotené a pripomienkované v peer review procese. Schválené príspevky budú nakoniec odprezentované.

Druhý typ konferencie sú uzavreté konferencie. Vtedy organizátori pozývajú jednotlivých prednášajúcich. Zväčša sa jedná o známych odborníkov, kde by kvalita príspevku mala byť zaručená. Tí poväčšine zasielajú iba abstrakt článku k schváleniu. Konferencie sú jednou z možností, ktoré kombinujú oba prístupy, kedy pozvaní významní odborníci dodávajú udalostí prestíž a predstavujú takzvaných kľúčových prednášajúcich. Konferenčné príspevky bývajú kratšie, niekoľkostranové, maximálne pár desiatok strán. Prednášajúci prezentujú svoje výsledky dvoma spôsobmi. Najskôr prednesú svoju prezentáciu v rozsahu desať až dvadsaťpäť minút, potom odpovedajú na päť až sedem otázok z publika a tiež prebieha diskusia k danej téme. Formát prezentovaných príspevkov môže mať predpísaný charakter v prípade, že budú príspevky knižne vydané. Podľa veľkosti rozdelujeme konferencie na multi track a single track. Single track sú menšie konferencie, pri ktorých sa prednášky neprekrývajú a bývajú prezentované v jednej prednáškovej miestnosti alebo budove. Multi track konferencie sa skladajú z niekoľko vzájomne prebiehajúcich prednášok. Môžu byť rozmiestnené vo viacerých budovách alebo v rôznych geografických lokalitách. Podľa rozsahu tém môžeme vedecké konferencie rozdeliť na tri nasledujúce typy:

- Tematické konferencie, menšie konferencie, ktoré sa venujú jednej konkrétnej téme.
- Všeobecné konferencie, konferencie so širším záberom tém. Bývajú zväčša organizované regionálnymi alebo medzinárodnými vedeckými spoločnosťami a konané pravidelne.
- Profesionálne konferencie, veľké konferencie, ktoré nie sú určené iba pre vedcov

a odborníkov, ale týkajú sa odborných tém. Väčšinou ide o konferencie typu multi track a súčasťou bývajú aj sprievodné akcie.

1.2.2 Príprava vedeckej konferencie

Organizácia konferencie, vytvorenie zborníku, či systému pre úspešnú konferenciu je zložitý proces, ktorý zahŕňa široké spektrum aspektov. Každý takýto prebieh sa môže od seba viac či menej líšiť, no príprava vždy zahŕňa nasledujúce časti.

- Organizačná skupina (ang. PC chair), pripraví najzákladnejšie informácie týkajúce sa konferencie. Zahŕňajú názov, témy konferencie, dátumy jednotlivých prípravných fáz, čas a miesto konania. Určia sa podmienky účasti a typ konferencie.
- Ďalšou časťou je výber jednotlivých odborných expertov (ang. Program Committee Member), ktorí budú zodpovední za peer review jednotlivých príspevkov. V ďalšom texte ich budem označovať aj ako recenzentov. Súčasťou býva aj hodnotiaci skupina odborníkov (ang. Senior Program Committee Member, Meta-Reviewer), ktorí dohliadajú nad celkovou kvalitou hodnotiaceho procesu a na základe odporúčaní recenzentov majú posledné slovo v schvaľovaní príspevkov.
- Následne nastáva fáza zasielania príspevkov jednotlivých účastníkov konferencie. Môže mať dve časti. V prvej časti autori zasielajú niektoré požadované informácie o príspevku zahŕňujúce názov a abstrakt. Na základe týchto informácií a abstraktu príspevku sú priradení recenzenti. V druhej časti autor zašle celý príspevok.
- Počas priradovacej fázy jednotliví recenzenti na základe abstraktu vyjadrujú svoje preferencie recenzovať konkrétny príspevok. Organizačná skupina priradí recenzentov k príspevkom tak, aby bolo zaručené, že príspevok hodnotí recenzent, ktorý je odborník v danej oblasti. Do úvahy sa pri rozhodovaní berie množstvo faktorov ako výsledky z priradovacej fázy, možné konflikty záujmov atď.
- Vo vyhodnocovacej fáze sa určí finálne rozhodnutie, ktoré príspevky budú schválené a ktoré zamietnuté. Vyhodnocovanie môže zahŕňať aj diskusiu nad výslednou formou článku medzi autorom a recenzentom. Obe strany vystupujú zväčša anonymne.
- Autori zasielajú finálnu verziu príspevkov. Táto forma môže mať predpísaný charakter v prípade, že príspevky budú aj knižne publikované.
- Organizačná príprava samotného priebehu konferencie predstavuje rozoslanie pozvánok, rezerváciu a rozvrhovanie prednáškových miestností, výber ubytovania pre účastníkov. A tiež zahŕňa prípravu sprievodných akcií napr. návrh

spomienkových predmetov ako trička, klúčenky atď.[24] [25]

1.3 Konferenčné systémy

Základom našej práce sú konferenčné systémy, preto si predstavíme, aké možnosti nám poskytujú. CMS sú webové aplikácie ktoré nám v podstatnom množstve uľahčujú celkovú organizáciu konferencií. Prvé CMS sa objavovali už v polovici deväťdesiatich rokoch, samozrejme s veľmi základnými možnosťami použitia, kedy podporovali iba komunikáciu účastníkov cez emaily. No s postupným vyvíjaním internetu sa vyvíjali aj CMS. V dnešnej dobe sú na trhu už vo veľkej miere komplexné konferenčné systémy. Umožňujú nám registráciu účastníkov konferencie ako napríklad recenzentov. Tak isto pridávanie príspevkov a ich následné uloženie do databázy, či export v rozličných formátoch. Takže nám každý takýto soft-ware slúži hlavne pre zber dokumentov v elektronickej forme a ich následné uloženie do úložiska, či databázy. Takto pozbierané dokumenty môžeme upravovať, triediť, alebo vyhľadávať. V našom prípade sú tieto dokumenty príspevky od autorov. Ďalšie možnosti, ktoré nám pomáhajú pri tvorbe konferencií si predstavíme v nasledujúcich častiach práce. Momentálne sa vo svete vyskytujú CMS vo veľkom množstve, či už komerčných, alebo open-source 1.3.1 alternatív. CMS nám neumožňujú iba prípravu konferencie, ale tak isto ho môžeme použiť aj pre prípravu vedeckých časopisov, alebo pri tvorbe zborníkov, ktoré budeme v našej práci vytvárať aj my. V podstate sú CMS použiteľné všade tam, kde je potrebný peer-review proces 1.3.2.

1.3.1 Open-source

Open source môžeme voľne preložiť ako otvorený software. Tento pojem používame pri aplikáciách ktoré majú otvorený zdrojový kód. Otvorený v tomto prípade znamená technickú dostupnosť kódu, ale aj legálnu dostupnosť. Táto nám pri dodržaní určitých podmienok umožňuje zdrojový kód využívať, prehliadať, alebo upravovať podľa potreby. [19]

1.3.2 Peer review

Slovo „peer“ vyjadruje odbornú rovnocennosť posudzovateľa s autorom príspevku, článku. Peer review (recenzia či posúdenie) je proces hodnotenia autorovej vedeckej práce, výskumu, alebo myšlienky za účelom zlepšiť prácu.[20]

1.4 Výber konferenčného systému

V práci sme vyhotovili stručný prehľad konferenčných systémov. Pretože každý CMS poskytuje aj veľa odlišných možností pre tvorbu konferencie, porovnali sme pár známych komerčných, ale aj open source systémov a ich hlavných funkcií a podľa nám zadaných kritérií pre tvorbu konferencie sme vybrali najvhodnejší z nich.

1.4.1 Prehľad konferenčných systémov

Konferenčný systém **COMS 13** je software vytvorený v Rakúsku. Ide o tzv. hostovský konferenčný systém, teda organizátor má nemôže použiť vlastný server pre údaje a tak isto sa o celú konferenciu starajú tvorcovia programu. Tiež pre to je vhodný pre usporadúvanie veľkých konferencií, minimálne so stopäťdesiatimi členmi. Aplikácia nám poskytuje jednu verziu programu, ktorá je na ich stránke voľne prístupná v demo verzii programu. Pre plnú verziu je tento systém platený. Ceny sú dostupné na stránke COMS a pohybujú sa od 150 až po 1000 eur, podľa počtu členov. Tak isto nám poskytuje generovanie konferenčných brožúr v rôznych formátoch, či vytvorenie stránky na mieru. Týmto systémom sme sa v práci viac nezaoberali, keďže našim cieľom je organizovať konferenciu s oveľa menším počtom členov.

Ďalej sme sa zaoberali konferenčným systémom **OpenConf 17**. OpenConf je považovaný za veľmi úspešný CMS, je používaný pre tvorbu konferencií po celom svete. Ako jeden z mála nevznikol na akademickej pôde, ale jeho tvorcom je spoločnosť Zákon Group. Zameriava sa hlavne na recenzentskú fázu. Je vhodný aj na vydávanie vedeckých časopisov. CMS poskytuje organizátorom tri verzie programu. Jedna verzia, komunitná, je verzia pre ľudí, ktorý sa uspokojia s licenčnými podmienkami a je dostupná zadarmo. Jej možnosti sú obmedzené. Organizátori musia disponovať vlastnými prostriedkami pre chod systému. Musia si zabezpečiť vlastný server, technikov. V tejto licenci nemožno používať technickú podporu iných firiem.

Druhá verzia, profesionál, už ponúka omnoho viac možností pre organizáciu a chod konferencie. Spoločnosť ponúka možnosť poskytnutia ich serveru. Ďalej stiahnutie zdrojových údajov, či možnosť meniť program na mieru. Tak isto v tejto verzii je poskytnutá online technická podpora. Cena poskytnutia tejto verzie sa pohybuje od sedemsto dolárov za štandardnú licenciu až do tisíc dolárov za Branding-Free licenciu 1.4.1.

Tretia verzia, plus, je v hlavných možnostiach zhodná s verziou profesionál. Vo verzii plus nemôžeme očakávať v prípade potreby server spoločnosti. Cena sa pohybuje od dvestopäťdesiat dolárov až po štyristopäťdesiat dolárov za Branding-free licenciu.

Branding-Free licencia umožňuje odstránenie všetkých autorských práv a ďalších

odkazov zo systému OpenConf.

EasyChair 16 je Konferenčný program vytvorený Andrejom Voronkovom z University of Manchester. Poskytuje nám dve verzie programu. Jedna je vhodná pre klasické konferencie tvorené jedným organizačným tímom a jednou skupinou recenzentov. Výsledkom je buď zborník, alebo konferencia. Druhá verzia nám ponúka vytvorenie takzvanej multi-track konferencie. Dokáže automaticky produkovať formát potrebný pre uverejňovanie do vedeckých časopisov. Tiež disponujú svojim vlastným L^AT_EX štýlom. Nevýhodou EasyChairu je, že je to čisto iba hosťovský konferenčný systém, teda program aj dáta musia byť uložené na serveroch univerzity. Za to výhodou je, že poskytuje obe verzie zadarmo, čo by nám v našom prípade vyhovovalo.

Conftool 15 je veľmi vyspelý konferenčný systém, vytvorený v Nemecku na Habsburskej univerzite. Systém môžeme zaradiť medzi systémy ktoré kladú dômysel na webovú prezentáciu, pričom ponúkajú aj online podporu. Je dostupný v dvoch verziách. Klasická verzia, ktorá má názov VSIS Conftool je šírená ako open/shared-soure aplikácia. Je zdarma pre organizovanie menších konferencií, pričom počet členov nesmie presiahnuť hranicu stopäťdesiat. Podporuje konferencie tvorené na akademickej pôde, pričom by mali poskytovať aj študentské zľavy.

Conftool pro je druhá verzia programu. Je prispôsobená organizácii väčších konferencií s členmi nad stopäťdesiat. Ponúka omnoho väčšiu škálu pokročilých funkcií, technickú podporu. Cena takejto vylepšenej verzie nebola na ich stránke zverejnená a závisí na celkovej veľkosti a počtu členov konferencie. Výhodou je aj česká a slovenská lokalizácia, čo ale v našom prípade nehrá veľkú rolu. Conftool poskytuje na stránke veľké množstvo návodov a pomôcok pre lepšie orientovanie v programe.

Konferenčný systém **Confious** 14, samotný tvorcovia označili ako pokrokový, s tým, že sa snažili odstrániť všetky nedostatky už existujúcich CMS. Systém patrí medzi hosťovské. Zabezpečuje automatické priradovanie recenzentov k daným článkom, disponuje detekciou konfliktov záujmov, čo je jeho veľmi zaujímavá vlastnosť.

1.4.2 Výber konferenčného systému

Pri výbere vhodného CMS sme sa chceli zamerať hlavne na systémy ktoré neposkytujú len hosťovské rozhranie. Tak isto by mal poskytovať dostupné a potrebné informácie pre export do nami vytvorených šablón vo viacerých formátoch. Dostupný konferenčný systém s technickou podporou a s možnosťou prispôbiť ho naším účelom. Finančnú stránku verzie programov sme veľmi nebrali do úvahy, keďže nami organizovaná konferencia nám poskytne finančné zdroje. Po preštudovaní vyššie spomenutých konferenčných systémov 1.4.1 sme zistili, že našim požiadavkám vyhovujú tri CMS.

Prvým z nich je konferenčný systém Conftool 1.4.1 . Tento CMS sa zdal z počiatku veľmi lukratívny a obsahuje naozaj veľa nám potrebných vlastností s dobrým prevedením a prezentáciu na webe. Podarilo sa nám nadviazať komunikáciu s ich podporou, na ktorej bolo vidieť ochotu a skúsenosti s usporadúvaním konferencií. Tento konferenčný systém sme sa rozhodli vylúčiť z nášho výberu pre vysoké finančné nároky. Aj keď sme v predchádzajúcej časti písali, že finančná stránka CMS nieje pre nás až taká podstatná, požadovali poplatok vo výške \$430 za základnú verziu, ktorá nám nepostačovala a hlavne v nej nebola zahrnutá technická podpora, ktorú budeme v ďalšej časti potrebovať. Pre nás potrebné vlastnosti by podľa informácií, ktoré sme od nich dostali, vyzdvihli cenu až do výšky \$1000, čo už predstavuje naozaj vysokú čiastku.

Druhým adeptom bol pre nás konferenčný systém Easychair 1.4.1 . Easychair sme ohodnotili ako veľmi prospešný software, ktorý naozaj dokáže ponúknuť mnoho zaujímavých možností. Väčšinu z nich sme popísali vyššie. Bohu žiaľ sa nám nepodarilo nadviazať komunikáciu s ich tímom aj po dlhšej dobe. Preto, že technická podpora CMS je pre nás dôležitá a ponúkaný software nám neposkytol ani odpovede na naše otázky, sme sa rozhodli tento systém vylúčiť z výberu.

Pri študovaní systému Confious sme zistili, že na ich stránke majú konferencie organizované Confiousom, avšak posledná bola konaná v roku 2010. Preto sme o poskytovaný systém neprejavili záujem.

Konferenčný systém OpenConf 1.4.1 sme zaradili na prvé miesto nášho výberu. Komunikáciu s ich podporou sa nám podarilo nadviazať naozaj veľmi rýchlo, čo považujeme za jeden zo znakov záujmu z ich strany. Pri komunikácii sme sa dozvedeli veľa dôležitých informácií, ktoré nám uľahčili výber. Ako sme už v predošlej časti spomínali, OpenConf poskytuje tri rôzne verzie softwaru. My sme sa rozhodli pre verziu „adition Plus“. Táto verzia síce neposkytuje všetky možnosti ktoré OpenConf ponúka, no sú pre nás postačujúce. Verzia Plus ponúka inštaláciu softwaru na náš vlastný server, čo je pre nás vyhovujúce. Možnosť vlastného serveru obsahuje mnoho výhod, ako aj menšia editácia programu, či ľubovoľná práca s databázou registrovaných účastníkov. Tiež nám táto verzia zabezpečuje už mnohokrát spomínanú technickú podporu, pridávanie abstraktov online alebo upload súborov až do veľkosti 64MB. Podporuje automatické či manuálne priradovanie recenzentov a tak isto obsahuje aj možnosť priradenia recenzenta od editora príspevku. Veľkou výhodou je automatická detekcia konfliktov, pri nahraní viac zhodných príspevkov. Zakomponovaná je aj automatická notifikácia recenzentom, pri nahraní potrebných súborov. Pre nás je podstatná možnosť exportu údajov v rôznych formátoch ako napríklad XML, CSV, excel, či SQL. Tieto hlavné vlastnosti konferenčného systému OpenConf nás viedli k výberu práve jeho, pričom cena za verziu Plus je \$250, čo je už veľmi akceptovateľné.

1.5 Konferencia DAFx



Obr. 1.1: DAFX

Naša práca má napomôcť k celkovej organizácii konferencie DAFX. V nasledujúcich riadkoch si priblížime o akú konferenciu sa jedná a čomu sa DAFX venuje. DAFX je synonymom pre digitálne zvukové efekty. Je to tiež meno pre európsky výskumný projekt pre spoluprácu a vedecký prenos, a to EÚ – COST – G6 „Digital Audio Effects“ (1997–2001). Projekt bol iniciovaný Danielom Arfibom (CNRS, Francúzsko). V posledných rokoch sme mali niekoľko medzinárodných seminárov / konferencií na DAFX. Ďalšia medzinárodná konferencia DAFX sa konala v roku 2014 v Erlangene. Paleta tém DAFX bola v priebehu týchto konferencií prezentovaná medzinárodnými účastníkmi. Príspevky možno nájsť na príslušných webových stránkach alebo v archíve.(DAFx.de) [3]

Pri opise vedeckých konferencií 1.2 sme sa dozvedeli, že vedecké konferencie môžu byť otvorené, alebo zatvorené. Nami organizovaná konferencia DAFX, je konferenciou, ktorá spĺňa obe tieto podmienky. Teda záujemcovia o prezentovanie posielajú svoje príspevky, poväčšine skrátené formy, na hodnotenie, či môžu byť odprezentované na konferencii. Avšak na konferencii DAFX budú pozvaní aj odborníci k danej téme, kde by kvalita príspevkov a prezentácii mala byť zaručená. Títo odborníci tak isto zväčšujú prestíž celému daniu konferencie. Očakávaný počet prijatých príspevkov na konferenciu je okolo osemdesiat, pričom prezentujúci nie sú výhradne z Českej republiky, ale takmer z celého sveta. Konferencia je typu single track. Teda celkový program diania bude spracovaný tak, aby sa jednotlivé prednášky neprekrývali, ale naopak chceme, aby na seba nadväzovali. Takýto priebeh zabezpečíme, ak prednášky budú odprezentované v jednej prednáškovej miestnosti, či nanajvýš v jednej budove. Ak zabezpečíme spomínaný priebeh, môžeme označiť program konferencie za plnárny. To znamená, že prezentácie budú konané za prítomnosti všetkých registrovaných účastníkov konferencie. Ďalej označujeme konferenciu DAFX ako konferenciu všeobecnú. DAFX sa síce venuje jednej konkrétnej téme a to digitálnym zvukovým

efektom, ale zároveň každý pozvaný, registrovaný prednášajúci sa zaoberá konkrétnou témou, ktorá súvisí so zvukovými efektami. Môžeme povedať, že konferencia sa zaoberá širšiemu záberu tém v okolí digitálnych zvukových efektov. Opisovaná konferencia je organizovaná pravidelne, každý rok a to od roku 1998, pričom sa vždy koná v inom štáte. Hlavným organizátorom je vedecká spoločnosť DAFx. Pre všetky tieto podnety teda zaraďujeme konferenciu do typu všeobecná.

Celkový program konferencie sa bude skladať z programu technického, kde sú zaradené prezentácie daných príspevkov, či diskusie na danú tému. A z programu spoločenského, ktorý zabezpečí organizátor konferencie. Pod spoločenský program môže byť zahrnuté slávnostné otvorenie konferencie, banket, či rôzne výlety, ku ktorým sa účastníci môžu prihlásiť pri registrácii na konferenciu. Technická podpora konferencie začína u webovej stránky, ktorá bude založená približne rok pred zahájením konferencie. V našom prípade nám bude webová stránka pridelená od organizácie DAFx, ktorú ďalej modifikujeme podľa našich potrieb. Na webovej stránke bude možnosť zaregistrovať sa cez príslušný konferenčný systém. Tak isto budú môcť účastníci, ktorí chcú na konferencii prednášať, cez webovú stránku a cez CMS poslať svoje príspevky do systému. K tým budú ďalej priradení recenzenti, ktorí sú určený z organizácie DAFx, aby daný príspevok ohodnotili. Ku každému príspevku budú priradený dvaja až traja recenzenti. Tí svoje hodnotenia odošlú cez web do systému. Ďalšie organizačné predmety sú takmer samozrejmosťou pre väčšinu konferencií. Myslíme tým napríklad účastnícku kartu konferencie, ktorá bude obsahovať meno, inštitúciu, v ktorej daný účastník pôsobí a štát. Túto kartu dostane každý registrovaný účastník pri vstupe na konferenciu. Ďalej sú samozrejmosťou darčkové predmety ako perá, prívesky, tričká a iné. Každý účastník tak isto dostane mapu Brna, brožúru s programom v cudzom jazyku. Preferuje sa anglický jazyk. Tak isto dostanú vygenerovaný zborník konferencie na USB, či za príplatok zborník v knižnej forme, čo si môže každý zvoliť pri registrácii na konferenciu.

2 VYUŽÍVANÉ TECHNOLOGIE

2.1 Jazyk PHP

PHP (Hypertext Preprocessor) je populárny open source skriptovací jazyk, ktorý sa používa najmä na programovanie klient-server aplikácií (na strane servera) a pre vývoj dynamických webových stránok. PHP bol inšpirovaný od rozličných programovacích jazykov, no najviac pobralo z jazyka C a jazyka Perl. Dokáže spolupracovať s relačnými databázami, ako napríklad MySQL, Oracle a podobne, pričom si vždy zanechá jednoduchú syntax. Beží na najrozšírenejších operačných systémoch ako sú Windows, UNIX, Linux, či Mac OS X. Je špeciálne navrhnutý na tvorbu webových stránok. Tiež môže byť využitý na skriptovanie z príkazového riadku. PHP zabezpečuje výstupný prúd dát vo forme najčastejšie HTML.

V našej práci s ním uvažujeme pri vytváraní správneho prostredia pre webovú stránku konferencie, export databázových údajov a vytvorenie správneho XML dokumentu. Táto možnosť však nieje výhodná.[21]

2.2 Jazyk XML

Zkratka XML, ako sme si už v úvode spomínali zastupuje anglický názov jazyka eXtensible Markup Language, čo preložíme ako „rozšírený značkovací jazyk“. Tento jazyk umožňuje pomocou tagov (značiek) označovať časti textu, ktoré môžeme ďalej rôzne využívať. Jazyk XML nieje ale jediný značkovací jazyk. Za veľikána môžeme označiť jazyk SGML (Standard Generalized Markup Language), ktorý patrí medzi prvé značkovacie jazyky. Umožnil definíciu vlastných značkovacích jazykov, na základe vlastnej definície typu dokumentu DTD. V roku 1986 bol jazyk SGML prijatý ako norma ISO. Bol ale príliš zložitý a vo svojej kategórii nenašiel uplatnenie. Bol tak isto predchodcom veľmi známeho jazyka HTML. Ten je veľmi jednoduchý a využíva sa hlavne pre vytváranie webových stránok. Tento jazyk je veľmi používaný a má mnoho výhod a nevýhod. Čím viac sa používal, tým vznikalo viac použiteľných značiek. Jeden z problémov jazyka HTML je, že veľa používaných tagov nepodporujú niektoré prehliadače, takže sa môže stať, že sa stránky zobrazia na rozličných prehliadačoch rozhádzane. HTML je jeden z dôvodov pre vznik jazyka XML. Tento jazyk má mnoho výhod. Jednou z nich je vlastnosť, že nemá žiadne predurčené tagy. Tým sa riešia hneď dva problémy iných jazykov. Nielen požiadavok pre minimálne množstvo značiek, ale aj možnosť mať značiek neobmedzený počet. XML má veľmi jednoduché syntaktické pravidlá a dbá na ich splnení. To je ďalšia z jeho výhod, ktorá je prospešná pre väčšinu vývojárov. Navyše nieje tak úzko spätý z angličtinou

a zahŕňa takmer všetky svetové jazyky. XML vznikol v rámci konsorcia W3C. Od svojho vzniku mal stanovených niekoľko vlastností, ktoré musel podporovať:

- Priame použitie v internete - prevzal niekoľko vlastností z HTML, aby sa mu to podarilo čo najľahšie.
- Podpora širokého spektra aplikácií - poskytuje vhodné prostredie pre popis rôznych druhov dát, či už databázové, správu dokumentu, alebo publikačné materiály.
- Kompatibilita s SGML – jazyk XML vznikol z SGML, takže všetky jeho vlastnosti vychádzajú práve z tohoto koreňového jazyka.
- Čo najnižší počet možných voliteľných vlastností – v tomto jazyku sa málo kedy stretáme s dvoma riešeniami jedného cieľa.
- Priehľadnosť a ľahká čitateľnosť – je založený na textovom formáte, takže je čitateľný pre všetkých užívateľov. Navyše všetky značky musia byť zapísané v kompletnej forme, takže nedochádza ku skráteným tvarom, čo opäť vyzdvihuje jeho jednoduchosť.
- Formálnosť a stručnosť návrhov – je definovaný pomocou veľmi jednoduchých pravidiel.
- Ľahká tvorba XML dokumentov – vytvárať dokumenty môžeme pomocou ľubovoľného textového editora.

V reálnom živote sa s ním stretáme hlavne na miestach, kedy chceme nejaký druh informácií publikovať v rôznych dokumentoch, či rôznych formátoch. Raz trebárs ako internetovú stránku, neskôr ako tlačенú knihu. S použitím XML, nemusíme vytvárať rôzne druhy dokumentov. Stačí, ak máme informácie uložené vo formáte XML a pomocou vhodných štýlov vygenerujeme požadované údaje. Alebo ak chceme určité informácie poskytnúť len nejakej skupine užívateľov, tak isto pomocou štýlov dosiahneme požadovaného cieľa.[27] [8] [28]

2.3 Definícia typu dokumentu DTD

Aby sme vytvorili správny XML dokument, musíme dbať na pravidlá tvorby. XML nám ponúka veľké množstvo pravidiel pre vytvorenie určitého typu dokumentu. Tomuto súhrnu pravidiel hovoríme Definícia Typu Dokumentu (Document Type Definition) DTD. Túto dôležitú vlastnosť jazyk XML zdedil z jazyka SGML. DTD nám umožňuje definovať vlastné značky (elementy, či atribúty). Tie sú ďalej k dispozícii pre vytvárané XML dokumenty podľa tohoto DTD. Pomocou DTD môžeme tiež určiť, akú majú mať elementy či atribúty hodnotu, aký majú obsah, alebo kde v dokumente sa nachádzajú. Jeho nevýhodou je, že samotný dokument DTD nieje XML súbor. Preto sa používajú aj ďalšie jazyky, ktoré určujú určité pravidlá, ako napríklad

XML Schema, alebo DSD, ktoré už sami o sebe sú vo formáte XML.[8] Pri vytváraní DTD musíme tiež postupovať podľa nejakých predom určených pravidiel. Prvý z definovaných elementov musí byť vždy element koreňový, teda element, ktorý v sebe obsahuje všetky ostatné prvky. Väčšinou sa však jedná o ďalšie elementy (potomky). Vtedy určíme pomocou znakov „+, ?, *“ koľkokrát sa dané elementy v koreňovom nachádzajú. Následne je potrebné definovať každý jeden element z koreňového elementov a to podľa toho, aké dáta v sebe obsahuje. Môžeme použiť:

#PCDATA, ANY, #CDATA

Definícia elementov musí začínať vždy značkou

```
<!ELEMENT ... ..>
```

kde na prvom prázdnom mieste sa nachádza meno elementu a na druhom jeho hodnota. Po definovaní elementov prichádza na rad definovanie príslušných použitých atribútov. V praxi sa môže stať že dokument neobsahuje žiadny atribút. No v prípade že nejaký obsahuje, mali by sme ho definovať hneď za elementom, v ktorom sa nachádza aby sme predišli chybám. Definícia atribútu musí vždy začínať značkou

```
<!ATTLIST ... ..>
```

za ktorou sa na prvom mieste sa nachádza názov elementu, do ktorého daný atribút patrí, potom nasleduje názov atribútu. Ďalej nasleduje hodnota, ktorá určí aký obsah atribút nesie. Na poslednom mieste určíme povinnosť atribútu. Ak je použitý popísaný základný postup pri tvorbe DTD, hovoríme, že je dokument validný a splňuje určitú úroveň správnosti.

2.4 Jazyk CSS

Ako sme už v úvode popísali, výhodou jazyku XML je oddelenie obsahu od grafického vzhľadu dokumentu. Preto, aby sme vytvorili dokument s vhodným obsahom aj vzhľadom, používame štýlové jazyky, ktoré umožňujú definovať konečný vizuálny vzhľad dokumentu. Pri použití s XML existuje viac druhov štýlových jazykov. Jedným z najjednoduchších jazykov tohoto typu je kaskádový štýl CSS (Cascading Style Sheets). Ten bol vytvorený predovšetkým pre potreby vývojárov HTML, určitý výsledný vzhľad webových stránok. CSS umožňuje určiť pre jednotlivé prvky, či skupiny elementov ich vlastnosti zobrazenia. Jazyk CSS sa vyvíjal a v ďalších verziách boli vytvorené vlastnosti určené priamo pre jazyk XML. Štýly CSS majú pomerne jednoduchú syntax a tak isto existujú pravidlá pre tvorbu takéhoto dokumentu. Zápis určenia vzhľadu musí obsahovať položku, ktorá sa nazýva selektor. Ten nám

určuje pre aký element, alebo skupinu elementov má byť definovaný vzhľad použitý. Selektor píšeme vždy na nový riadok. Musíme dbať na to, aby sme používali presné názvy zhodné s názvami v dokumente XML. Samotná definícia vzhľadu je zapísaná za selektorom vždy v zložených zátvorkách a musí obsahovať minimálne jednu deklaráciu. Deklarácia sa skladá vždy z vlastnosti a hodnoty, ktorú chceme priradiť. Vlastnosti a hodnoty oddeľujeme od seba dvojbodkou. Teda napríklad pomocou definície:

```
table{background-color : red};
```

určíme že tabuľka bude mať pozadie červenej farby. Výhodou je, že CSS môžeme zapísať do dokumentu XML, či HTML, ale tak isto ho môžeme pripojiť ako externý súbor. To má za následok zmenšenie dátovej veľkosti dokumentu. Naopak nevýhodou je, že CSS nezaručuje rovnaké vykresľovanie vo všetkých prehliadačoch.[8] [2]

2.5 Jazyk XSL

XSL (Extensible Stylesheet Language) je jazyk ktorý nám napomáha k určovaniu zobrazenia dokumentu XML. Na rozdiel od kaskádových štýlov ale vznikol priamo v súvislosti s jazykom XML, takže nám ponúka oveľa viac možností ako spomínaný kaskádový štýl CSS. Jazyk CSS neumožňuje trebárs preskupovať alebo triediť elementy, či meniť obsah dokumentu a vykonávať výpočty. V súvislosti s XSL spomenieme aj jazyk DSSSL (Document Style Semantics and Specification Language), ktorý bol vytvorený pre použitie v SGML. Bol ale natoľko zložitý, že nenašiel príliš veľké uplatnenie. Jazyk XSL vznikol preto, aby odstránil nedostatky CSS, ale zároveň nebol až taký zložitý ako jazyk DSSSL. Môžeme o ňom hovoriť ako o „zlatej strednej ceste“. Výhodou tohoto jazyka je, že využíva syntax jazyku XML, takže každý kto vytvára takýto štýl dokumentu, sa nemusí učiť nový jazyk. XSL je bol vytvorený konzorciom W3C a skladá sa z dvoch častí:

- XSLT-metóda pre transformáciu XML dokumentov.
- XSLFO-metóda formátovania XML dokumentov.

Jazyk **XSLT** umožňuje meniť jeden dokument XML na iný dokument XML s odlišným obsahom alebo usporiadaním, na dokumenty podporujúce mobilné telefóny, či iné formáty. Dokáže triediť a filtrovať dáta, či opakovane používať prvky zo zdrojového kódu. Môžeme ho využiť aj pri vytváraní dokumentu PDF použitím PDF \LaTeX u. Najčastejšie sa však XSL používa na prevod XML dokumentu do dokumentu HTML. XSLT je v dnešnej dobe najdôležitejšia aj najpoužívanejšia časť štandardu XSL. Pri použití jazyka XSLT je potrebné mať príslušný procesor pre vyhotovenie výstupného dokumentu zo vstupného dokumentu XML. Ak chceme vytvoriť XSLT dokument, je nutné aby odpovedal všetkým syntaktickým pravidlám

pre tvorbu XML, pretože sa vlastne jedná o dokument typu XML. V praxi môžeme XSLT využiť práve tam, kde je potrebné častejšie meniť obsah stránky. Vytvoríme si XSLT dokument, ktorý bude transformovať dokument z XML. Následne pri každej zmene obsahu nám stačí už len upraviť údaje v dokumente XML, pričom XSLT necháme zachovaný a automaticky vygenerujeme nám potrebnú stránku HTML.

Jazyk **XSLFO** ponúka možnosti podobné s jazykom CSS. Veľa jeho príkazov sa dokonca s vlastnosťami CSS zhoduje. Popisujeme ho ako prezentačný jazyk. Neobsahuje žiadne sémantické značkovanie ako HTML. Na rozdiel od CSS, ktorý upravuje prezentáciu externého XML alebo HTML dokumentu, v sebe obsahuje všetky dáta dokumentu [9] [10] [11] .

2.6 MySQL

MySQL je slobodný a otvorený viacvláknový, viac užívateľský SQL relačný databázový server. Veľkou výhodou je, že je podporovaný viacerými platformami a implementovaný viacerými programovacími jazykmi ako napríklad C++, PHP či Perl. Databáza v MySQL je tvorená potrebným množstvom tabuliek, ktoré sú rozdelené do stĺpcov udávajúcich dátový typ záznamu a riadkov v ktorých sú samotné záznamy uložené. Jednotlivé záznamy sa do tabuliek databázy vkladajú a vyberajú dotazmi jazyka SQL. MySQL databáza sa vo veľkom spája s tvorbou webu a webových aplikácií. Konferenčný systém OpenConf 1.4.1 disponuje práve popisovaným databázovým systémom MySQL. Preto v našej práci využijeme MySQL na prácu so záznamami jednotlivých účastníkov konferencie a tak isto na vytváranie potrebného XML súboru. Môžeme povedať, že MySQL tvorí veľmi dôležitú časť našej práce. [4]

2.7 L^AT_EX

L^AT_EX je vysoko kvalitný typografický systém určený pre profesionálne a poloprofesionálne sádzanie dokumentov. Bol vyvinutý v roku 1985 Leslie Lamportom. Mohli by sme povedať, že sa jedná o balík makier programu T_EX. Tento balík umožňuje autorom osádzať text vo veľmi vysokej kvalite. Práca v L^AT_EXu spočíva v tom, že do textu článku, alebo práce sa vkladajú formátovacie príkazy. Výhodou je, že autor si tak môže sám zdefinovať potrebný formát strany priamo pri písaní daného textu. V našej práci využijeme L^AT_EX práve na kvalitné spracovanie zborníka konferencie, kedy použijeme potrebné formátovacie príkazy tohoto programu, ktoré je možné zapísať do dokumentu XSL pri vytváraní šablóny pre zborník a následne pomocou nástroja PDFL^AT_EX vytvoríme správne formátovaný cieľ práce. [23]

3 POPISOVANÉ TECHNOLOGIE POUŽITÉ V PRAXI

3.1 Použitie PHP

Ako sme si povedali, PHP je jazyk, ktorý poskytuje množstvo nástrojov pre prácu s webom a webovým prostredím, s databázami rôznych typov a tak isto na komunikáciu so serverom. V našej práci tento jazyk využijeme hlavne pri doplňovaní a upravovaní webovej stránky konferencie, ktorú nám poskytnú organizátori konferencie DAFx. Ďalej sme PHP využili pre nadviazanie spojenia s databázou konferenčného systému, pre získavanie informácií obsiahnutých v potrebnom XML súbore. Konferenčný systém disponuje databázou MySQL. Pre nadviazanie spojenia s MySQL databázou sme vytvorili jednoduchý program, ktorý sa pripojí na databázu, automaticky sa prepojí do potrebnej databázovej tabuľky a umožní nám pracovať s obsahujúcimi dátami. Následne sme vytvorili PHP kód, ktorý nám potrebné informácie z databázovej tabuľky zapíše do nami požadovaných elementov v dokumente XML. Týmto sme zabezpečili, že XML súbor bude vytvorený presne podľa našich potrieb a iba s informáciami, ktoré skutočne potrebujeme. Teda, predídeme tak zbytočnej neprehľadnosti dokumentu.

3.2 Použitie jazyka XML v našej práci

V našej práci využijeme možnosti jazyka XML v mnohých smeroch a pre všetky nami požadované výsledky práce. Ako sme si už povedali do XML môžeme ukladať rôzne druhy informácií a tak isto môžeme tieto informácie publikovať rôznymi spôsobmi. Keď sme opisovali nami pripravovanú konferenciu, zmienili sme sa s tým, že každý registrovaný dostane pri vstupe na konferenciu účastnícku kartu s určitými údajmi. Pri vytváraní týchto kariet nám tak isto pomôže jazyk XML. Každý účastník pri registrácii na konferenciu vypíše registračný formulár, pomocou zvoleného konferenčného systému, ktorý bude obsahovať pre nás dôležité informácie. Tieto informácie sa následne uložia do databázy systému. Z tejto databázy, ako sme už spomínali, vytvoríme potrebný XML súbor. Výhodou je, že XML súbor stačí vytvoriť iba jeden a všetky informácie zapísané v tomto súbore nám budú k dispozícii pri vytváraní všetkých požadovaných cieľov práce. Pri vytváraní XML sme postupovali nasledovne. Najskôr sme si určili, ktoré informácie sú pre nás potrebné a na základe toho sme vytvorili koreňový element s názvom *members*. Tento element v sebe zahŕňa všetkých príslušných prednášajúcich účastníkov konferencie. Pritom

je každý jeden účastník zaznamenaný v elemente *user*, ktorý nesie identické číslo zaznamenané v atribúte *id*. Pomocou tohoto *id* je možné každého *user*-a identifikovať a použiť jeho údaje samostatne pre rozličné operácie potrebné pre splnenie našich cieľov. Samozrejme informácie nemusíme používať len samostatne, ale aj spoločne zadaním príkazov celkovo pre všetky elementy *user*. Element *user* obsahuje ďalšie podelementy a to *card* a *data*. Element *card* nám poskytuje všetky potrebné informácie pre vytvorenie účastníckych kariet, teda meno, priezvisko, krajinu z ktorej daný prednášajúci pochádza a organizáciu v ktorej pôsobí. Avšak aby sme predišli komplikovanosti XML súboru, tieto dáta ako meno a priezvisko tiež využijeme ďalej pri vytváraní PDF zborníka a HTML stránky s príspevkami daných prednášajúcich. Element *data* v sebe zahŕňa podobne ako element *user*, identifikačný atribút tentokrát *id1*, pomocou ktorého zabezpečíme, že každý užívateľ môže pridať viac ako jeden príspevok a následne je možné tieto príspevky zobraziť na potrebných miestach spolu s menom daného autora, nezávisle od seba. Spomínaný element *data* v sebe obsahuje informácie ako sú názov súboru v preddefinovanom formáte PDF a nadpis príspevku, ktorý bude prednášaný. Všetky údaje obsiahnuté v elemente *data* nám výrazne prispievajú pri vytváraní zborníka. Do tejto časti sme chceli zahrnúť aj element, ktorý by niesol informácie o ceste k uloženým súborom, avšak nakoniec sme sa rozhodli tieto údaje do XML nezahrnúť, pretože cesta k uloženým príspevkom je vždy rovnaká a teda je výhodnejšie ju zadať do programu staticky. Časť kódu súboru môžeme vidieť nižšie 3.2. Pomocou rôznych príkazov a za pomoci štýlu XSL potom určíme, kde sa majú dané údaje zobrazovať a tým určíme výsledný vzhľad účastníckych kariet, zborníka či HTML stránky. Samozrejme teraz neberieme do ohľadu grafickú úpravu. Tú budeme rozoberať nižšie.

```
<members>
  <user id="28">
    <card>
      <name>Aaron</name>
      <sc_name>Wishnick</sc_name>
      <country>Massachusetts, USA</country>
      <organisation>iZotope, Inc.,</organisation>
    </card>
    <data id1="1">
      <file_name>dafx14_aaron_wishnick_time_varying_.pdf</file_name>
      <title>TIME-VARYING FILTERS FOR MUSICAL APPLICATIONS</title>
    </data>
    <data id1="2">
      <file_name>dafx14_aaron_wishnick_second_presentation_.pdf</file_name>
```

```
        <title>SECOND_PRESENTATION_OF_WISHNICK</title>
    </data>
</user>
</members>
```

3.3 Použitie DTD

O vlastnostiach DTD sme si už povedali pár dôležitých informácií v predchádzajúcej kapitole. V našej práci využijeme DTD pre definíciu všetkých elementov určujúcich osobné údaje účastníkov konferencie, ale aj údaje určujúce názvy súborov, či atribúty priradené daným elementom. Tie sme mohli vidieť v predchádzajúcej časti 3.2. V kóde sa nachádza koreňový element *members*, ktorý deklarujeme ako prvý a zapíšeme mu hodnotu obsahu elementu „user“ s tým, že tento element sa u neho vyskytuje viac ako jedenkrát. Následne zapíšeme hodnotu elementu *user* tak, že obsahuje ďalšie podelementy *card* a *data*. Opäť sme k elementu *data* pripísali znak „+“, ktorý nám určuje, že tento element sa v rodičovskom elemente, teda v *user*-ovi vyskytuje minimálne jedenkrát, pričom maximálny počet nieje obmedzený. Pri definícii *card* pripisujeme hodnotu „?“ a teda, že tento element sa v rodičovskom elemente nachádza maximálne raz.

Je dôležité, aby pri definovaní atribútov v dokumente sme ich definíciu zaradili priamo za definíciu elementu, ktorému patrí. Preto sme ďalším krokom definovali atribút *id*. Táto definícia obsahuje názov elementu, ku ktorému je atribút priradený, v našom prípade teda *user*. Aké dáta obsahuje, my sme vybrali CDATA, pretože môže obsahovať akýkoľvek reťazec, v našej práci čísla. A nakoniec sme mu priradili hodnotu *#REQUIRED* ktorá znamená, že tento atribút je povinný. Následne sme definovali element *card* a to tak, že obsahuje podelementy *name*, *sname*, *country* a *organisation*. Tiež je dôležité ako aj v prípade atribútov, aby sme podelementy definovali za elementom, v ktorom sa nachádzajú. Teda sme týmto elementom priradili hodnoty *#PCDATA*, čo znamená, že obsahujú textové údaje. Podobne ako *card* sme zložke *data* stanovili obsahujúce podelementy a následne sme deklarovali atribút *id1* ako v prípade *user*-a a podelementy *filename* a *title*. Týmto sme sa uistili a zabezpečili, že XML súbor má správnu štruktúru a je vhodný pre použitie v našej práci. Rovnako ako u XML aj DTD dokument je postačujúce vytvoriť iba raz, nakoľko sme si povedali, že XML súbor sa nemení počas celej našej práce, tak DTD dokument sa nemení tiež. Avšak nemôžeme zabudnúť na to, že DTD dokument musíme ešte pripojiť do XML súboru, ktorému je určený. A to tak, že ho zapíšeme do hlavičky XML dokumentu, ako môžeme vidieť na obrázku 3.1.

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

Obr. 3.1: DTD v hlavičke dokumentu XML

Následne si môžeme pozrieť popisovaný DTD súbor.

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT members (user+)>
<!ELEMENT user (card?, data+)>
<!ATTLIST user id CDATA #REQUIRED>
<!ELEMENT card (name, sc_name, country, organisation)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT sc_name (#PCDATA)>
<!ELEMENT country (#PCDATA)>
<!ELEMENT organisation (#PCDATA)>
<!ELEMENT data (file_name, title)>
<!ATTLIST data id1 CDATA #REQUIRED>
<!ELEMENT file_name (#PCDATA)>
<!ELEMENT title (#PCDATA)>
```

3.4 Použitie jazyka XSL

Jazyk XSL obsahuje množstvo vlastností, ktoré sú vhodné pre vytvorenie upravených XML dokumentov. Vela z týchto vlastností sme si opísali v predchádzajúcej kapitole 2.5. V našej práci tak isto našiel tento jazyk uplatnenie. Konkrétne sme doposiaľ využili jeho časť XSLT, konkrétnejšie XSL–Transformáciu. Zatiaľ sme v našej práci popisovali vytvárané dokumenty, ktoré sa používajú rovnako pre všetky vytvárané ciele. Z tohto pohľadu môžeme použitie jazyka XSL v našej práci rozdeliť na tri časti, pretože XSL je potrebné vytvárať pre každú časť samostatne.

- XSL pre vytvorenie účastníckych kariet
- XSL pre vytvorenie HTML zborníka
- XSL pre vytvorenie PDF zborníka

3.4.1 XSL pre automatizované vytváranie účastníckych kariet

Účastnícka karta, ktorá je požadovaný cieľ práce, nieje na pohľad zložitá. No pri vytváraní automatizovanej generácií tejto karty sme museli dbať na striktné umiestnenie parametrov, či elementov, aby sme predišli možnému chaosu v kartách v zmysle, že by mohla byť každá takáto karta iná, respektíve by mohlo nastať ľubovoľné rozmiestnenie údajov zakomponovaných v nej. Preto sme sa po podrobnejšom preštudovaní XSL a XSLT rozhodli formát účastníckej karty vytvoriť pomocou tabuliek. Pri vytváraní správneho XSL dokumentu je prvoradá správne zadefinovať hlavičku 3.2 dokumentu spolu so správnym kódovaním, ktoré musí byť zhodné pre všetky dokumenty, teda XML, DTD, XSL a samozrejme aj s databázou MySQL. My sme sa rozhodli pre kódovanie UTF-8.

```
<?xml version='1.0' encoding='UTF-8' ?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                version="2.0">
<xsl:output method="text" encoding="utf-8"
            omit-xml-declaration="yes"/>
```

Obr. 3.2: Hlavička XSL dokumentu

Následne sme sa prepli do elementu v ktorom chceme vykonávať zmeny, teda u nás to je koreňový element *members*. Učinili sme tak pomocou príkazu

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

ktorý je zapísaný hneď na začiatku dokumentu a platí až kým ho neukončíme uzatváracím tagom `</xsl:template>`, no spravíme tak až ku koncu dokumentu. V dokumente XSL je potrebné vytvoriť, čo sa má ďalej v koreňovom elemente *members* diať. Ako sme popisovali XML dokument v predchádzajúcich kapitolách 3.2, vieme že element *user* sa nachádza v koreňovom elemente viackrát ako jeden a preto musíme zabezpečiť aby sa vkladané príkazy použili pre všetkých účastníkov, teda pre všetky elementy *user*. Príkazom

```
<xsl:for-each select="user">
```

určíme práve popisujúcu funkciu. Všetky príkazy napísané pod nim, sa budú používať pre každého *user*-a až kým tento príkaz neukončíme ukončovacím tagom. Teraz prichádza na rad samotne vytvorenie tabulky s potrebnými údajmi. Keďže je nutné účastnícke karty generovať vo výstupnom formáte PDF, použili sme preto možnosť generovania pomocou PDF_{La}TeXu, ktorý je popísaný nižšie 3.4.3, avšak v prostredí

jazyka XSL vkladáme potrebné príkazy pre túto generáciu. Ako sme sa rozhodli formát účastníckych kariet vytvárať pomocou tabuľky, túto tabuľku vložíme pomocou príkazov pre L^AT_EXa nie príkazmi XSL. Do tabuľky následne vložíme príkaz ktorý do danej bunky v tabuľke vloží danú hodnotu z dokumentu XML. Toto sme uskutočnili nasledujúcimi príkazmi:

```
<xsl:apply-templates select="card/name"/><xsl:text> </xsl:text>
<xsl:apply-templates select="card/sc_name"/>
<xsl:apply-templates select="card/country"/>
<xsl:apply-templates select="card/organisation"/>
```

Pre funkčnosť dokumentu sme nakoniec ošetrili, čo sa ma do daných buniek z elementov vypísať. Zvolili sme element *name* a pred samotný obsah elementu sme vložili popis „Name“ pre úpravu a lepší finálny vzhľad účastníckych kariet.

```
<xsl:template match="name">
  <xsl:text>Name: </xsl:text>
  <xsl:apply-templates/>
</xsl:template>
```

Rovnakým spôsobom sme postupovali aj pri všetkých ostatných potrebných elementoch v tabuľke. Ak by bolo potrebné podľa organizátorov konferencie pridať do kariet aj logo konferencie, využijeme pre tento účel tiež šablónu XSL. V prípade potreby je možné do dokumentu XSL vložiť súbor CSS v ktorom môžeme mať tiež definované ako má účastnícka karta vyzeráť, či akú má mať farbu a pod.

3.4.2 Automatizované vytváranie HTML stránky príspevkov

Vytvorenie HTML zborníku konferencie je jeden z uvedených cieľov práce. Pri realizovaní tohoto cieľu opäť použijeme vytvorený XML súbor 3.2, DTD dokument 3.3, no XSL dokument je potrebné vytvoriť samostatne. Preto sme preštudovali problematiku vytvárania HTML stránky za pomoci týchto spomínaných dokumentov a rozhodli sa pre riešenie, ktoré sa čo najviac približuje zadaniu práce. Pri vytváraní nám pomohla inšpirácia z predošlých ročníkov konferencie, ako by mala táto stránka približne vyzeráť. Obdobne ako pri riešení problematiky účastníckych kariet sme ako prvé zadefinovali správne kódovanie celého dokumentu v hlavičke XSL a tiež sme vytváraný XSL súbor zapísali do hlavičky XML dokumentu 3.3, aby sme predišli komplikáciám.

Opäť musíme určiť v ktorom elemente XML dokumentu budeme vykonávať operácie. Pre náš potrebný výstup znova zvolíme koreňový element *members*. Aby sme

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE vizitka SYSTEM "card.dtd">
<?xml-stylesheet type="text/xsl" href="html_proceedings.xsl"?>

```

Obr. 3.3: Hlavička dokumentu XML

zabezpečili výstupný výzor stránky, použijeme príkazy používané pre tvorbu webu pomocou HTML. XSL využíva pre tvorbu tieto príkazy, čo nám uľahčuje vytváranie HTML zborníka. Hlavné nadpisy stránky sme obsadili v bunkách tabuľky, ktorú sme potrebné naformátovали a zarovnali, pre správne zobrazenie. Tieto príkazy musia byť umiestnené za výberom koreňového elementu, ale zároveň pred ostatnými príkazmi pre prácu s ďalšími elementami. A to preto, lebo koreňový element, ako sme si už spomínali, sa v dokumente nachádza práve jedenkrát, teda aj hlavné nastavenia a formát bude zobrazený práve jedenkrát. Následne pokračujeme v definícii príkazov pre ďalšie elementy dokumentu. Zobrazovanie mena príslušného prezentujúceho účastníka zabezpečíme pomocou príkazov :

```

<xsl:template match="card">
  <p>
    <xsl:apply-templates select="name"/>
    <xsl:apply-templates select="sc_name"/>
  </p>
</xsl:template>

```

Teda, dokument sme prepli do elementu *card* a zvolili sme jeho podelementy s ktorými sa budú vykonávať operácie. Následne určíme, aké operácie sa majú s týmito elementami vykonať. Nám ide hlavne o vypísanie mena a priezviska autora príspevku. Pred vypísaním vytvoríme formátovateľný objekt `` v ktorom budú tieto údaje vypísané a zvolíme si hodnoty zobrazenia. V našom prípade používame jednoduché zobrazenie tučným písmom pre zvýraznenie údajov.

```

<xsl:template match="name">
  <span>
    <b>
      <xsl:value-of select="."/>
    </b>
    <xsl:text> </xsl:text>
  </span>
</xsl:template>

```


Obdobne sme postupovali pri vkladaní priezviska, pre dosiahnutie rovnakého efektu. Pre vytvorenie HTML zborníku musíme ešte zabezpečiť správne otváranie uložených príspevkov. Na to využijeme podelementy elementu *data*, ktoré však musíme ošetriť správnymi XSL príkazmi. Poslúži nám na to hlavne vytvorenie atribútu. XSL nedisponuje všetkými príkazmi pre tvorbu webu ako HTML, ale umožňuje nám si niektoré príkazy vytvoriť práve spomínaným vytvorením atribútu. Preto sme zapísali do dokumentu tag `<a>` a práve jemu priradili atribút `href`, ktorý slúži na otváranie odkazov. K platnému použitiu musíme tomu atribútu ešte priradiť hodnotu, respektíve cestu k súboru, ktorý má byť otvorený. Túto cestu sme zapísali staticky `./upload/`, pretože je nemenná, s tým že doplnený názov súboru, ktorý má byť otvorený pochádza opäť z XML dokumentu. Je potrebné si uvedomiť, že použité príkazy treba ukončiť uzatváracím tagom na správnom mieste príkazu. K úspešnému vytvoreniu atribútu `href` musíme ešte priložiť príslušný text cez ktorý, po kliknutí sa nám otvorí požadovaný príspevok. Na to nám bude slúžiť element *title* ako nadpis príspevku. Opísané vytváranie atribútu môžeme vidieť nižšie.

```
<xsl:template match="data">
  <a>
    <xsl:attribute name="href">./upload/<xsl:value-of select="file_name"/>
  </xsl:attribute>
    <span><xsl:value-of select="title"/></span>
  </a>
</xsl:template>
```

3.4.3 XSL pre vytvorenie automatizovaného PDF zborníka

Vytvorenie PDF zborníka konferencie DAFx je hlavný cieľ našej práce, ktorý umožní takto vytvorený zborník ďalej prezentovať ako súbor na internetovej stránke konferencie. Ďalej vytvorený PDF zborník použijeme pre tlač knihovej verzie zborníka konferencie. Pre uskutočnenie týchto cieľov sme sa inšpirovali tlačeným zborníkom vytvoreným pre konferenciu DAFx 2014. Pomocou tohoto zborníka sme si vytvorili požadovanú štruktúru, uviedli aké časti má tlačený zborník obsahovať, uviedli sme si, ktoré časti zborníka majú byť automatizované pomocou XML, DTD a XSL súborov a ktoré časti majú byť v zborníku zaznamenané staticky, teda napevno. K tomuto účelu nám poslúžil nástroj pre vytváranie PDF zo spomínaných dokumentov, PDF \LaTeX . Veľké množstvo príkazov v XSL dokumente pre vytvorenie zborníka, slúži hlavne pre program \LaTeX a pre umožnenie transformácie týchto dokumentov práve pomocou PDF \LaTeX u. My sa v tejto kapitole zameriame ako aj na príkazy XSL, tak aj na príkazy programu \LaTeX u. Pretože príkazy sa navzájom prekrývajú

a v niektorých prípadoch sú na sebe závislé, nebolo by vhodné tvorbu tohoto cieľa rozdeliť do viac častí.

Hlavnou otázkou bolo, ako vytvoriť program, ktorý bude pridané príspevky prezentujúcich užívateľov zoradovať v určitom zmysle za sebou, vytvorí tak naformátovanú postupnosť PDF súborov. Opäť sme uskutočnili dôležité kroky, ako pri tvorbe predchádzajúcich XSL dokumentov a to precízne ošetriť hlavičku dokumentu, teda kódovanie a pridelenie XSL dokumentu do hlavičky XML súboru. Tiež sme na začiatku dokumentu určili v ktorom elemente sa budú vykonávať operácie, teda opäť koreňový element *members*. Potom prichádzajú na rad príkazy pre L^AT_EX a to veľmi podstatné, teda vkladanie usepackage-ov. Konkrétnejšie si tieto esepackage objasníme pri použití. Je veľmi dôležité uvedomiť si, že príkazy pre L^AT_EX sa nestotožňujú s príkazmi pre XSL. Preto ak chceme vkladať takéto príkazy do dokumentu XSL musíme daný text uviesť medzi XSL tagy `<xsl:text>` a `</xsl:text>`, čím zabezpečíme program proti chybám v XSL kóde. Usepackage

```
\usepackage{fancyhdr}
```

nám umožňuje definovať požadovanú hlavičku strán dokumentu a tiež jej parametre ako sú veľkosť písma, či rám hlavičky. Preto je nezbytné, aby bol vložený do šablóny. Ďalší veľmi dôležitý usepackage je

```
\usepackage[overlay]{textpos}
```

ktorý nám umožňuje vytvoriť textové bloky, ktorým môžeme priradiť presné umiestnenie na príslušnej strane dokumentu podľa vertikálnej a horizontálnej hladiny. Volba *overlay* nám zaručuje, že daný text sa nebude prekrývať so žiadnym iným textom v prostredí. Príkaz textových blokov potom jednoducho zavoláme pomocou

```
\begin{textblock}{10}(0,12)
```

pričom za textom musí byť ukončený. V dokumente často používame usporiadanie odborníkov pomocou tabuliek. Pri vytváraní tabuliek sme si v úvode dokumentu zadefinovali vlastnú bunku „L“. Výhodou je že si pri definovaní bunky môžeme dopredu zabezpečiť potrebný formát, veľkosť bunky, či zarovnanie. Všetko nám dovoľuje príkaz

```
\newcolumntype{L}{ >{\raggedright\arraybackslash} m{7.5cm} }
```

Pri vytváraní tabuliek je dôležité upozorniť, že PDF^LA_TE_X pomocou XSL nám neumožňuje oddelovať bunky klasickým znakom `&`, ale je nutné použiť zápis znaku `&`. Ostatný statický text sme pridávali pomocou klasických príkazov L^AT_EXu. Cez statický text sme sa dostali až k časti zborníku, ktorý má byť automaticky

generovaný pomocou XML dát a príkazov XSL, teda ku vkladaniu príslušných príspevkov. Dvojicou príkazov

```
<xsl:for-each select="user">
  <xsl:for-each select="data">
```

určíme, že pre každý element *user*, ktorých je veľké množstvo, a následne pre každý element *data*, ktorý môže byť v každom *user*-ovi obsiahnutý viac ako jedenkrát, budú platiť príkazy uvedené medzi týmto tagom a uzatváracím tagom `</xsl:for-each>`. V nich tiež uvádzame príkazy \LaTeX u, ako sú príkaz pre vloženie príslušného príspevku do obsahu zborníka, ktorý musí byť sprevádzaný príkazom pre fiktívnu sekciu, aby fungovalo hypertextové prepojenie s obsahom a tak isto musí byť zadaný usepackage, ktorý umožňuje túto operáciu.

```
\usepackage{tocloft}
\phantomsection
  \addcontentsline{toc}{subsection}{}
}
```

V prázdnej zátvore je uvedený text, ktorý sa má v obsahu zobrazíť. My sme použili príkazy pre vkladanie mena, priezviska a názvu príspevku z XML dokumentu. Teda príkazy

```
<xsl:value-of select="../card/name"/>
<xsl:value-of select="../card/sc_name"/>
</xsl:text><xsl:value-of select="title"/>
```

ktoré nám zapíšu hodnotu z príslušného elementu. Nasleduje dvojica príkazov pre vloženie daného PDF súboru za pomoci usepackage `pdfpages`.

```
\shorthandoff{-}
  \includepdf [pages=-] {}
}
```

Príkaz `includepdf` obsahuje parameter `pages`, ktorého hodnota určuje ktoré strany z daného PDF dokumentu majú byť zobrazené. My sme v našej práci zvolili „-“, čo znamená, že zobrazíť sa majú všetky strany dokumentu. Tento parameter je nutné pred jeho použitím zdefinovať pomocou `shorthandoff-`. V prázdnych zátvorkách sa nachádza cesta k týmto požadovaným súborom, ktorú sme obdobne ako v predchádzajúcom prípade 3.4.2 zapísali staticky s tým, že názov otváraného súboru je zapísaný pomocou XSL príkazu

```
<xsl:apply-templates select="file_name"/> .
```

Ten má v ďalšej časti kódu definované čo má vlastne robiť. V našom prípade vypísať obsah elementu. Teraz môžeme príkazy ukončiť uzatváracím tagom `</xsl:for-each>`. Tým sme zabezpečili, že pre každého prezentujúceho konferencie, budú všetky jeho príspevky vložené do zborníka k jeho menu a zároveň bude každý tento príspevok vložený do obsahu zborníka. Opísaný kód môžeme vidieť na obrázku nižšie 3.4 .

```
<xsl:for-each select="user">
<xsl:for-each select="data">
  <xsl:text>
    \phantomsection
    \addcontentsline{toc}{subsection}{\\ \textbf{</xsl:text>
<xsl:value-of select="../card/name"/>
<xsl:value-of select="../card/sc_name"/>
<xsl:text>}}\\
    \scriptsize{</xsl:text><xsl:value-of select="title"/>
    <xsl:text>}}
    \shorthandoff{-}
  \includepdf[pages=-]{upload/</xsl:text><xsl:apply-templates
    select="file_name"/>
<xsl:text>}
</xsl:text>
</xsl:for-each>
</xsl:for-each>
```

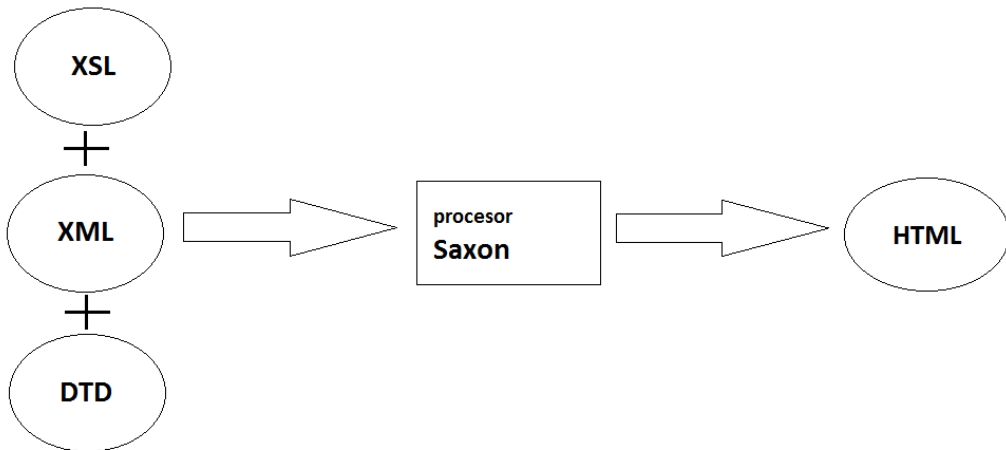
Obr. 3.4: Príkazy pre automatické vkladanie príspevkov a obsahu

3.5 Výsledky práce pomocou procesoru Saxon

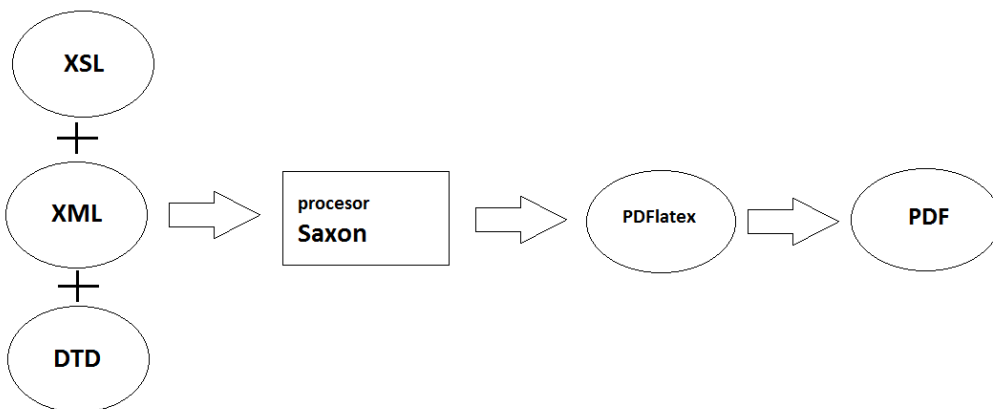
Ako sme už v predchádzajúcich častiach spomínali, na vytvorenie každého požadovaného výsledku práce sú potrebné dokumenty DTD, XML a XSL. Ich štruktúru a funkčnosť sme opísali vyššie 3 . Aby z takto vytvorených dokumentov vznikol požadovaný výsledok, je nutné uskutočniť XSL-Transformáciu. My v našej práci používame pre túto transformáciu procesor Saxon. Saxon je voľne šíriteľný XSLT procesor, ktorý nájdeme na stránkach softwaru 29 . Je dostupný v rôznych verziách, pričom najnovšie verzie poskytujú procesy pre XSLT 2.0. Saxon je vytvorený v prostredí Java, preto pre jeho spustenie je nutné mať na počítači nainštalovaný balík Java. Saxon pracuje v terminálovom okne a pre správne spracovanie dokumentov je nutné zadať vstupné a výstupné parametre, pre každý požadovaný cieľ. Parametre zadávame nasledovne:

```
java -cp . -jar saxon9he.jar -expand:on -o:<názov výstupného súboru>  
<názov XML>.xml <názov XSL>.xsl
```

Je dôležité zdôrazniť, že pokiaľ potrebné súbory pre transformáciu nie sú uložené v jednom adresári, musíme k parametrom zadať aj cestu k nim. Princíp práce Saxonu je znázornený na obrázku 3.6



Obr. 3.5: Vytvorenie HTML stránky procesorom Saxon

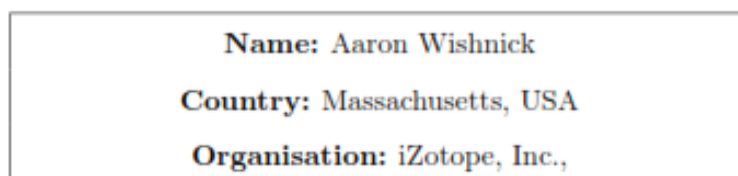


Obr. 3.6: Vytvorenie PDF súboru pomocou procesora Saxon

4 VÝSLEDKY ŠTUDENTSKEJ PRÁCE

4.1 Účastnícke karty

Aby sme vytvorili požadovaný výstup práce a teda účastnícke karty pomocou procesoru Saxon, musíme doň zadať správne vstupné parametre. Ako sme si povedali, XML súbor je pre všetky časti práce rovnaký, preto aj zápis jeho hodnoty do Saxonu je vždy rovnaká. Musíme ešte zadať názov XSL súboru, ktorý musí byť zhodný s názvom v XML hlavičke dokumentu. Účastnícke karty sa už vytvoria automaticky pre každého účastníka podľa zadaných príkazov v XSL, vo formáte a pripravené na tlač.



Obr. 4.1: Vytvorená účastnícka karta

4.2 HTML zborník konferencie

HTML zborník konferencie spočíva v tom, že pomocou vytvoreného XML súboru, XSL a DTD vytvoríme užitím procesora Saxon jednoduchú HTML webovú stránku, ktorá bude obsahovať mená a príspevky zaregistrovaných účinkujúcich konferencie. Táto stránka sa automaticky aktualizuje podľa záznamov v XML a teda nemôže dojsť k tomu, aby niektoré údaje chýbali. V XSL šablóne pre tvorbu tohoto dokumentu vložíme HTML príkazy pre jednoduchú úpravu a nadpisy. V prípade potreby je možno pomocou týchto príkazov vložiť na úvod stránky tiež logo konferencie. Vytvorenú HTML verziu zborníka napokon pridáme do stránky konferencie DAFx 2016, kde bude prístupná užívateľom. Pomocou správnych príkazov určíme Saxonu potrebné parametre pre vytvorenie HTML stránky a za spustenia terminálového okna, jednoducho vznikne HTML stránka, ktorá je spustiteľná v internetových prehliadačoch. Ukážka takto vytvorenej stránky je na obrázku 4.2.

DAFX 2016

Proceedings of the 19th International Conference on Digital Audio Effects

Aaron Wishnick

[TIME-VARYING FILTERS FOR MUSICAL APPLICATIONS](#)

[SECOND PRESENTATION OF WISHNICK](#)

Vesa Norilo

[EXPLORING THE VECTORED TIME VARIANT COMB FILTER](#)

Brian Hamilton

[FINITE DIFFERENCE SCHEMES ON HEXAGONAL GRIDS FOR THIN LINEAR PLATES WITH FINITE VOLUME BOUNDARIES](#)

Obr. 4.2: Časť HTML zborníka konferencie

4.3 PDF zborník konferencie

Vytvorenie zborníka konferencie vo formáte PDF je najzložitejší z našich požadovaných cieľov. Obsahuje veľké množstvo príkazov potrebných pre formátovanie pomocou PDF \LaTeX u, usepackage bez ktorých by príkazy neboli plne funkčné a tiež rad XSL príkazov pre vkladanie dát z odpovedajúceho XML dokumentu. Počtom a formátom strán sa približuje knižnej verzii zborníka na ktorú je predurčený. Vďaka spomenutým príkazom sa v zborníku nachádza automaticky generovaný obsah spolu s príspevkami určenými k tlači. PDF zborník tiež vytvoríme užitím procesora Saxon, kedy po zadaní správnych vstupných a výstupných parametrov vzniká spomínaný cieľ práce. Titulná strana zborníku je predbežná a môžeme ju zmeniť na základe požiadavok organizátorov konferencie.

5 ZÁVER

Hlavnými cieľmi bakalárskej práce bolo zistiť, aké existujú konferenčné systémy pre prípravu konferencie DAFx v roku 2016. A následne porovnať viacero takýchto systémov a podľa vlastností vybrať vhodného adepta. Konferenčný systém mal v sebe zahrňovať určité vlastnosti a to napríklad možnosť exportu databázových informácií v nám potrebných formátoch, či zahrňovať technickú podporu. Ďalej sme mali vytvoriť vhodnú xsl šablónu tak, aby automatizovane pomocou vstupného XML dokumentu a za pomoci potrebných nástrojov či procesov vznikli účastnícke karty pre tlač a zborník konferencie vo formáte PDF a ako HTML stránka.

Zadanie bakalárskej práce bolo splnené. V kapitole 1.4 sme spravili prehľad existujúcich známych konferenčných systémov, vzájomne sme ich porovnali a na základe zistených vlastností sme vybrali vhodný systém. Požadované vlastnosti spĺňa konferenčný systém OpenConf, s ktorým sme sa rozhodli naďalej pracovať. Zaoberali sme sa možnosťami jazyka XML a jeho použitím. Navrhovali sme vhodné doplnenie pomocou jazyka XML a jeho súčastí, ktoré sú popísané v kapitole 2. Pomocou jazyka PHP sme vytvorili XML dokument vhodný pre splnenie cieľov. Tiež sme vytvorili tri odlišné XSL dokumentu, ktoré nám automatizované generovanie podporných materiálov pre konferenciu umožnia. Použitím týchto šablón sme vygenerovali požadované ciele práce, teda správne naformátovaný zborník konferencie pomocou \LaTeX u zložený z príspevkov prednášajúcich vo formáte PDF, účastnícke karty pripravené na tlač a tiež zborník vo formáte HTML stránky.

Bakalárska práca môže byť ďalej použitá pre tvorbu konferenčných materiálov danej konferencie v roku 2016, za predpokladu použitia totožného konferenčného systému a tiež použitím rovnakých nástrojov a jazykov.

LITERATÚRA

- [1] Biskupská konferencia *charakteristika Biskupskej konferencie*[online]. Dostupné z URL: <http://cs.wikipedia.org/wiki/Biskupsk%C3%A1_konference>.
- [2] CSS *charakteristika štýlu CSS*[online]. Dostupné z URL: <<http://webdesign.about.com/od/beginningcss/a/aa021607.htm>>.
- [3] DAFx *popis DAFx inštitúcie*[online]. Dostupné z URL: <www.dafx.de>.
- [4] Databázový systém MySQL *Základná charakteristika MySQL*[online]. Dostupné z URL: <<http://techterms.com/definition/mysql>>.
- [5] Elektronická konferencia *charakteristika elektronickej konferencie*[online]. Dostupné z URL: <http://cs.wikipedia.org/wiki/Elektronick%C3%A1_konference>.
- [6] E-mailová konferencia *charakteristika E-mailovej konferencie*[online]. Dostupné z URL: <http://cs.wikipedia.org/wiki/E-mailov%C3%A1_konference>.
- [7] ESPOSITO, Dino. *XML: efektívne programovanie pro .NET. 1. vyd.* Praha: Grada, 2004, 596 s. ISBN 80-247-0775-6.
- [8] GRUSOVÁ, L. *XML pre úplných začiatčníkov* Praha: Computer Press, 2002. ISBN 80-7226-697-7
- [9] Jazyk XSL *základná charakteristika jazyku XSL v anglickom jazyku*[online]. Dostupné z URL: <<http://en.wikipedia.org/wiki/XSL>>.
- [10] Jazyk XSL *základná charakteristika jazyku XSL v českom jazyku*[online]. Dostupné z URL: <http://cs.wikipedia.org/wiki/Extensible_Stylesheet_Language>.
- [11] Jazyk XSLT *základná charakteristika jazyku XSLT v anglickom jazyku*[online]. Dostupné z URL: <<http://en.wikipedia.org/wiki/XSLT>>.
- [12] Konferencie *popis konferencií*[online]. Dostupné z URL: <<http://cs.wikipedia.org/wiki/Konference>>.
- [13] Konferenčný systém COMS *Domovská stránka systému*[online]. Dostupné z URL: <<http://www.conference-service.com/index.html>>.
- [14] Konferenčný systém Confious *Domovská stránka systému*[online]. Dostupné z URL: <<http://www.confious.com/>>.

- [15] Konferenčný systém Conftool *Domovská stránka systému*[online]. Dostupné z URL: <<http://www.conftool.net/>>.
- [16] Konferenčný systém Easychair *Domovská stránka systému*[online]. Dostupné z URL: <<http://www.easychair.org/>>.
- [17] Konferenčný systém Openconf *Domovská stránka systému*[online]. Dostupné z URL: <<http://www.openconf.com/>>.
- [18] KOSEK, Jiří. *PHP a XML. 1. vyd.* Praha: Grada, 2009, 367 s. ISBN 978-80-247-1116-4
- [19] Open Source software *charakteristika termínu open source* [online]. Dostupné z URL: <<http://www.abclinuxu.cz/slovník/open-source>>.
- [20] Peer review *charakteristika termínu peer review*[online]. Dostupné z URL: <<http://guides.lib.jjay.cuny.edu/content.php?pid=209679&sid=1746812>>.
- [21] PHP *základná charakteristika jazyku PHP*[online]. Dostupné z URL: <<http://webdesign.about.com/od/webprogrammingglossary/fl/php-defined.htm>>.
- [22] Tlacova konferencia *charakteristika tlacovej konferencie*[online]. Dostupné z URL: <http://cs.wikipedia.org/wiki/Tiskov%C3%A1_konference>.
- [23] Typografický systém \LaTeX *Popis \LaTeX* [online]. Dostupné z URL: <<http://sk.wikipedia.org/wiki/LaTeX>>.
- [24] Vedecká konferencia *popis vedeckej konferencie*[online]. Dostupné z URL: <http://rationalwiki.org/wiki/Scientific_conference>.
- [25] Vedecká konferencia *Popis vedeckej konferencie*[online]. Dostupné z URL: <http://is.muni.cz/th/99126/fi_m_b1/DIPLOMOVA-PRACA.txt>.
- [26] Williams, Hugh E. *PHP a MySQL : vytváříme webové databázové aplikace : podrobný průvodce tvůrce WWW stránek* Praha: Computer Press, 2002, 530 s. ISBN 80-7226-760-4.
- [27] XML *základná charakteristika jazyku XML*[online]. Dostupné z URL: <<http://sk.wikipedia.org/wiki/XML>>.
- [28] XML *charakteristika jazyku XML*[online]. Dostupné z URL: <<http://xbrleducation.com/edu/xml.htm>>.

[29] XSLT procesor Saxon *Domovská stránka systému*[online]. Dostupné z URL:
<<http://saxon.sourceforge.net/>>.

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

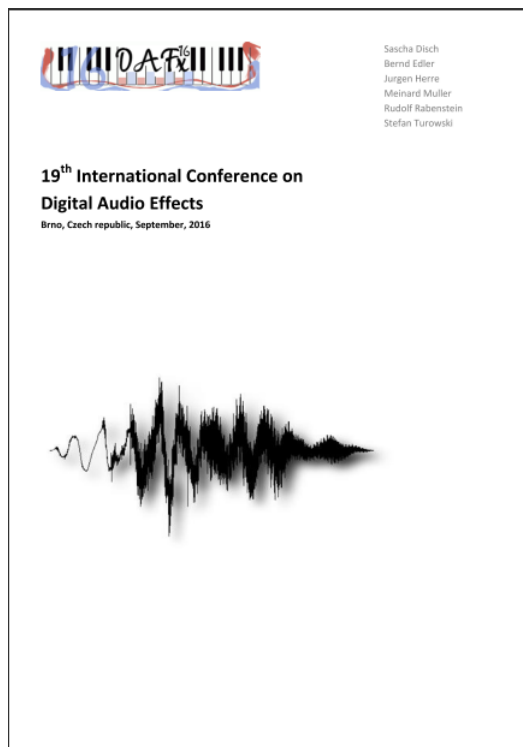
XML	rozšírený značkovací jazyk–eXtensibleMarkup Language
XSL	jazyk popisujúci transformáciu XML súborov–eXtensible Stylesheet Language
XSLT	transformácia na prevod zdrojových dát z XML do iného formátu–eXtensible Stylesheet Language Transformations
XSLFO	značkovací jazyk na bázy XML pre formátovanie dokumentov–eXtensible Stylesheet Language - Formatting Objects
LaTeX	system pre sádzanie dokumentov
CMS	konferenčný systém–Conference Managment Systems
PC Chair	organizačná skupina konferencie– Program Committee Chair
Meta-Reviewer	hodnotiaca skupina odborníkov
VSIS	Distribuované systémy a informačné systémy–Verteilte Systeme und Informationssysteme
MB	jednotka kapacity pamäťových médií–megabyte
CSV	jednoduchý súborový formát vo forme čistého textu–comma-separated values
SQL	štruktúrovaný vyhľadávací jazyk–Structured Query Language
DAFx	digitálne zvukové efekty–Digital Audio Effects
USB	univerzálna sériová zbernica–Universal Serial Bus
MySQL	slobodný a otvorený viacvláknový, viac užívateľský SQL relačný databázový server
HTML	Hypertextový značkový jazyk–HyperText Markup Language
C	štandardný programovací jazyk
C++	štandardný programovací jazyk
Perl	univerzálny, interpretovaný programovací jazyk–Practical Extraction and Report Language

SGML	všeobecný značkovací jazyk–Standard Generalized Markup Language
DTD	definícia typu dokumentu–Document Type Definition
ISO	medzinárodná organizácia pre normalizáciu– International Organization for Standardization
W3C	konzorcium produkujúce slobodné štandardy–World Wide Web Consortium
CSS	kaskádový štýl CSS–Cascading Style Sheets
DSSSL	štýlový dokument pre špecifikáciu jazyka–Document Style Semantics and Specification Language
PDF	súborový formát–Portable Document Format

ZOZNAM PRÍLOH

A Ukážka vytvoreného PDF zborníku	46
B Obsah priloženého CD	47

A UKÁŽKA VYTVORENÉHO PDF ZBORNÍKU



Obr. A.1: Obal zborníku konference

Proc. of the 19th Int. Conference on Digital Audio Effects (DAFx-16), Brno, Czech republic, September, 2016

Conference Committees

DAFx Board

Daniel Arfib
(CNRS-LMA, Marseille, France)

Sylvain Marchand
(LaBRI, Université Bordeaux 1, France)

Nicola Bernardini
(Conservatorio di Musica „Cesare Pollini“,
Padova, Italy)

Damian Murphy
(University of York, UK)

Francisco Javier Casajús
(ETSI Telecomunicación - Universidad
Politécnica de Madrid, Spain)

Soren Nielsen
(SoundFocus, Arhus, Denmark)

Laurent Daudet
(LAM / IJLRA, Université Pierre et Marie
Curie (Paris VI), France)

Markus Noisternig
(IRCAM, France)

Philippe Depalle
(McGill University, Montreal, Canada)

Luis Ortiz Berenguer
(EUIT Telecomunicación - Universidad
Politécnica de Madrid, Spain)

Giovanni De Poli
(CSC, University of Padova, Italy)

Geoffroy Peeters
(IRCAM, France)

Obr. A.2: Časť vytvoreného zborníka

B OBSAH PRILOŽENÉHO CD

Na priloženom CD sa nachádzajú štyri adresáre a to *cards*, *conference proceedings*, *html_proceedings* a *PHP*.

Adresár **cards** obsahuje všetky súbory potrebné pre vytvorenie vizitiek k tlačí vo formáte PDF.

- Súbor *card* v príkazovom riadku, teda skript, ktorý po spustení vytvorí pomocou procesoru Saxon požadovaný PDF výstup.
- Súbor *card* vo formáte DTD 3.3.
- Šablóna XSL pre vytvorenie vizitiek s názvom *card* a koncovkou *.xsl* 3.4.1.
- Už vytvorený PDF súbor s vizitkami *card_PDF*.
- XML súbor s názvom *data* 3.2
- Procesor Saxon.

Pre vygenerovanie vizitiek k tlačí je potrebné spustiť skript *card*, ktorý spustí procesor Saxon. Na tento úkon je potrebné mať v počítači nainštalovaný balík JAVA.

Adresár **conference proceedings** podobne ako predchádzajúci, tiež obsahuje všetky súbory potrebné pre vytvorenie PDF zborníka konferencie. Obsahuje:

- Adresár *logo*, v ktorom sa nachádza nami vytvorené logo konferencie.
- V adresári *obal* sa nachádzajú súbory pre zobrazenie obalu zborníka.
- Priečink *upload* v sebe zahŕňa všetky príspevky od daných účastníkov konferencie, ktoré majú byť zobrazené v zborníku.
- DTD dokument *card* pre správne definovaný XML súbor 3.3.
- *Contribution_PDF* je vytvorený zborník konferencie vo formáte PDF.
- Skript pre vytvorenie zborníku pomocou procesoru Saxon, *contribution*.
- *Contribution* XSL súbor je šablóna pre vytvorenie a sadzbu zborníka 3.4.3.
- XML súbor *data* s informáciami o účastníkoch konferencie 3.2.
- Procesor Saxon.

Ak chceme znova vygenerovať PDF zborník konferencie, je nutné spustiť v tomto priečinku súbor *contribution* a to aspoň dva krát pre správne vygenerovanie zborníka.

Adresár **html_proceedings** v sebe zahŕňa súbory pre vytvorenie zborníka konferencie v formáte HTML stránky. Obsahuje:

- Priečink *upload*, v ktorom sú uložené dané príspevky užívateľov, ktoré chceme zverejniť.
- DTD súbor *card* 3.3.
- XML súbor *data*, s potrebnými údajmi užívateľov 3.2.
- *html_proceedings* je vygenerovaný zborník konferencie vo formáte html.
- Skript pre generáciu html zboríka s názvom *pdf_page_prevod*.
- XSL dokument *proceedings*, teda šablóna pre vytvorenie tohoto zborníka 3.4.2.

- Procesor Saxon.

Pre opätovné vytvorenie html zborníka konferencie musíme najskôr odstrániť súbor *html_proceedings* a následne spustiť skript *pdf_page_prevod*, ktorý nám už sám vygeneruje html zborník konferencie.

Adresár **PHP** v sebe zahŕňa:

- PHP súbor *connect* pre pripojenie sa k danej databáze systému.
- PHP súbor *xmlcreate*, ktorý slúži pre vytvorenie XML súboru z údajov v danej databáze.

Popisované súbory v adresári *PHP* slúžia ako pred vytvorená šablóna, kde v prípade potreby zmeníme údaje databázy v súbore *connect* a pomocou *xmlcreate* vytvoríme potrebný XML súbor. Tieto PHP súbory nejde spustiť bez toho, aby boli pripojené k potrebnej databáze a tiež musia byť uložené na serveri. Teda v našom prípade nám slúžili ako pomôcka pri vytváraní požadovaných súborov a po modifikovaní a uložení na správnom mieste budú použité pri nadviazaní komunikácie s konferenčným systémom.