

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

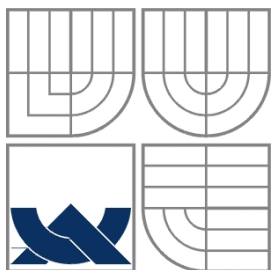
WEBOVÝ PROHLÍZEČ PŘEDNÁŠEK

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

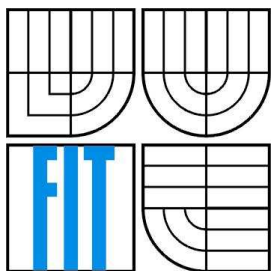
AUTOR PRÁCE
AUTHOR

Bc. JOSEF ŽIŽKA

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

WEBOVÝ PROHLÍZEČ PŘEDNÁŠEK

WEB-BASED LECTURE BROWSER

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. JOSEF ŽIŽKA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. MICHAL FAPŠO

BRNO 2009

Abstrakt

Tato práce se zabývá webovým prohlížečem přednášek, jehož cílem je usnadnit přístup k informacím s využitím moderních řečových a multimediálních technologií. Jsou rozebrány jednotlivé technologie, na kterých je prohlížeč postaven. Významnou roli v prohlížeči představují videozáznamy, a proto je velká část této práce zaměřena na digitální video a způsoby jeho doručení skrze streaming servery. Jsou uvedena řešení podobně stavěných multimediálních prohlížečů. Čtenář je seznámen s návrhem prohlížeče. Toto zahrnuje popis jeho jednotlivých komponent a je vysvětlena jejich vzájemná synchronizace. Je představena výsledná verze prohlížeče přednášek, jsou zmíněny problémy, které se vyskytly v průběhu jeho vývoje a při nasazení prohlížeče do reálného provozu. Na konci práce je diskutován budoucí vývoj webového prohlížeče přednášek.

Abstract

This thesis deals with a web-based lecture browser. Its goal is to facilitate the access to information with the use of modern speech and multimedia technologies. Technologies used for this browser are discussed. Video recordings play a very important role in the browser, and therefore the big portion of this work is aimed at the digital video and methods of its delivery using streaming servers. Solutions of similar multimedia browsers are mentioned. The reader is acquainted with the browser design. This includes describing the various components of the browser and how their mutual synchronization is done. The final version of the browser is introduced and the problems that occurred during the development process and deployment into service are mentioned. In the conclusion of this work the future development of the web-based lecture browser is discussed.

Klíčová slova

prohledávání audio dat, zpracování řeči, video streaming, multimediální prohlížeč, grafické uživatelské rozhraní, XML

Keywords

audio browsing, speech processing, video streaming, multimedia browser, graphical user interface, XML

Citace

Žižka Josef: Webový prohlížeč přednášek, diplomová práce, Brno, FIT VUT v Brně, 2009

Webový prohlížeč přednášek

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Michala Fapša. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Josef Žižka
22. dubna 2009

Poděkování

Rád bych v první řadě poděkoval panu Ing. Michalu Fapšovi za konzultace a podporu při řešení mé diplomové práce. Dále mé poděkování patří panu Ing. Petru Schwarzovi za zajímavé nápady. Velký dík náleží panu Ing. Vítězslavu Beranovi za implementaci systému pro automatické zpracování slajdů z videozáznamů přednášek. Za poskytnutí sady ukázkových dat a zodpovězení dotazů ohledně pořizování videozáznamů na FIT VUT v Brně děkuji paní Mgr. Janě Skokanové. Za instalaci serveru a vyřešení určitých technických otázek děkuji panu Ing. Tomáši Kašpárkovi. Rovněž děkuji členům výzkumné skupiny Speech@FIT, zejména pak jejímu vedoucímu Doc. Ing. Janu Černockému, a kamarádům, zvláště Ing. Martinu Krčovi, za jakýkoliv druh podpory v rámci řešení této práce.

© Josef Žižka, 2009

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Obsah.....	1
1 Úvod.....	3
2 Technologie.....	4
2.1 Video.....	4
2.1.1 Multimediální kontejner.....	5
2.1.2 Kodek.....	7
2.2 Video streaming.....	7
2.2.1 Windows Media Services.....	8
2.2.2 Microsoft SilverLight.....	8
2.2.3 Helix.....	8
2.2.4 QuickTime Streaming Server.....	8
2.2.5 VideoLAN streaming.....	9
2.2.6 Flash video streaming.....	10
2.3 Řečové technologie Speech@FIT.....	11
2.3.1 Rozpoznávání řeči s velkým slovníkem LVCSR.....	11
2.3.2 Detekce klíčových slov nebo frází.....	11
2.3.3 Identifikace a verifikace mluvěcího.....	12
2.3.4 Rozpoznávání jazyka.....	12
2.3.5 Indexování a vyhledávání v řečových datech.....	12
2.4 Zpracování slajdů z videozáznamů přednášek.....	14
2.4.1 Návrh základního systému.....	14
2.4.2 Implementovaný systém.....	16
2.4.3 Alternativní řešení.....	16
3 Prohlížeče přednášek.....	17
3.1 MBrowser.....	17
3.2 JFerret.....	17
3.3 MIT Lecture Browser.....	18
3.4 Videlectures.net.....	19
4 Realizace prohlížeče přednášek.....	20
4.1 Počáteční fáze.....	20
4.2 Cíle projektu.....	21
4.3 Architektura systému.....	21
4.4 Kategorizace obsahu.....	22
4.5 XML soubory.....	22
4.5.1 Jazykové rozhraní prohlížeče.....	23
4.5.2 Uložení dat přednášek.....	23
4.6 Komponenty prohlížeče.....	25
4.6.1 Videozáznam.....	25
4.6.2 Vyhledávání v audio datech.....	26
4.6.3 Transkript.....	26
4.6.4 Slajdy.....	28
4.6.5 Odkazy.....	29
4.6.6 Informace o přednášce.....	30
4.6.7 Časová osa přednášky.....	30
4.6.8 Příbuzné přednášky.....	31

4.7	Synchronizace komponent	31
4.8	Uživatelské rozhraní.....	31
4.9	Ukázková data	33
4.10	Technické řešení.....	34
4.10.1	Přidávání nových dat do prohlížeče.....	34
4.10.2	Použité aplikace třetí strany.....	35
5	Hodnocení	37
5.1	Možnosti širšího uplatnění prohlížeče.....	37
6	Rozšíření prohlížeče přednášek.....	38
6.1	Technické řešení prohlížeče	38
6.2	Vlastnosti prohlížeče	38
6.2.1	Inteligentní linkování obsahu.....	38
6.2.2	Uživatelsky definovaný prohlížeč.....	38
6.2.3	Oznámení o aktualizaci	39
6.2.4	Rozšíření typů videozáznamů přednášek	39
6.2.5	Online streaming.....	39
6.3	Návrh dalších komponent.....	39
6.3.1	Komentáře.....	40
6.3.2	Vylepšený transkript	40
6.3.3	Slajdy	40
6.3.4	Ostatní.....	40
7	Závěr	41
	Literatura	42
	Seznam příloh	44

1 Úvod

V dnešní době mají multimédia pro člověka velký význam. S postupným příchodem nejrůznějších technologií je člověku život v mnoha směrech stále více usnadňován. Cílem této práce je ulehčit uživatelům výsledného systému přístup k informacím s využitím moderních řečových a multimediálních technologií.

Představte si, že navštěvujete kurz týkající se operačních systémů. K dispozici jsou videozáznamy všech přednášek. Při studiu na zkoušku potřebujete pochopit funkčnost obecného semaforu. Jak ale rychle získat požadované vysvětlení, když máte k dispozici několik dvouhodinových přednášek? Většina videozáznamů přednášek sice obsahuje název, případně stručný popis, čeho se přednáška týká, nicméně i tak je pracné nalézt ono 10minutové správné místo uvnitř dvouhodinového videa. Přitom obecný semafor mohl být probrán i na jiných přednáškách, v dalších předmětech, či zmíněn na začátku, uprostřed nebo na konci přednášky. A takové hledání stojí člověka v lepším případě pouze čas. Toto dilema se snaží vyřešit systém pro indexování a vyhledávání v řečových datech, který je vyvíjen výzkumnou skupinou Speech@FIT. Díky tomuto systému a různým moderním technologiím může být studentům výrazně usnadněn přístup k požadovaným informacím. S touto myšlenkou také přichází webový prohlížeč přednášek, o kterém pojednává tato diplomová práce.

Následující kapitola 2 přináší čtenáři náhled do technologií, na kterých je webový prohlížeč přednášek vybudován. Důležitým prvkem prohlížeče přednášek je schopnost pracovat s videozáznamy. Z tohoto důvodu je podstatná část kapitoly zaměřena na technologie umožňující streaming videa a jeho nejčastější řešení. Jelikož prohlížeč využívá některé řečové technologie vyvíjené skupinou Speech@FIT, jejich přehled a stručný popis je rovněž zmíněn. Poslední část kapitoly je věnována zpracování slajdů z videozáznamů přednášek. Je navržen možný postup pro vytvoření časové osy s momenty, kdy dochází ke změnám obsahu promítacího plátna. Následně je představen implementovaný systém pro automatické zpracování slajdů z videozáznamů přednášek.

Kapitola 3 seznamuje čtenáře s podobně situovanými prohlížeči. Část kapitoly je věnována multimediálnímu prohlížeči MBrowser, který je vyvíjen na Fakultě informačních technologií VUT v Brně a jehož řešení spolu s dalšími uvedenými prohlížeči slouží jako hlavní zdroj inspirace pro tuto diplomovou práci.

Realizaci webového prohlížeče přednášek rozebírá kapitola 4. Je popsána počáteční fáze řešení projektu a jsou uvedeny cíle, které byly stanoveny. Je vysvětlen princip kategorizace obsahu, jsou podrobně popsány jednotlivé komponenty a je nastíněna jejich vzájemná synchronizace v módu přehrávání přednášky. Čtenář je rovněž obeznámen s XML soubory, které slouží jako hlavní úložiště pro data prohlížeče. Je rozebrán návrh webového rozhraní. Protože zpracování každého videozáznamu probíhá v několika krocích s využitím různých rozpoznávačů a aplikací, je uveden jejich popis a přínos pro prohlížeč přednášek. Je popsána architektura prohlížeče, jeho technické řešení a jsou uvedeny problémy, které bylo nutné při vývoji překonat. Množina ukázkových dat, kterou byl prohlížeč naplněn, je představena.

Kapitola 5 hodnotí vytvořený systém a uvažuje jeho použití i pro jiné účely než jsou univerzitní přednášky.

O navrhovaných rozšířeních pro prohlížeč přednášek pojednává kapitola 6. Tento seznam jednotlivých doporučení je důležitý zejména pro ty, kteří budou dále pokračovat v rozvoji tohoto projektu.

Myšlenkou této práce je tedy vytvoření reálně použitelného prohlížeče přednášek, který bude pomáhat uživatelům se vstřebáváním informací probíraných na přednáškách.

2 Technologie

Cílem této kapitoly je seznámit čtenáře s dostupnými technologiemi, na kterých může být postaven webový prohlížeč přednášek. Záměrem není popsat všechny technologie, nýbrž přiblížit zejména ty, které mohou mít pro prohlížeč přednášek speciální přínos a budou jej tak významně odlišovat od ostatních řešení. Jedná se zejména o využití různých systémů pro zpracování obrazu a mluveného slova.

Významným prvkem prohlížeče jsou videozáznamy, a proto je velká část této kapitoly zaměřena na digitální video a způsoby jeho doručení skrze streaming servery. Následuje přehled a stručný popis některých řečových technologií, které jsou vyvíjené skupinou Speech@FIT. Poslední část této kapitoly je zaměřena na automatickou extrakci slajdů z videozáznamů přednášek a její možný přínos pro prohlížeč přednášek.

2.1 Video

Slovo video pochází z latiny, kde znamená vidět. Jedná se o technologii pro zaznamenávání, přehrávání, přenos a obnovu pohyblivých obrázků používající elektronické signály nebo digitální média. Termín video se běžně vztahuje ke způsobům ukládání obrazových záznamů, které mohou být digitální (DVD, MPEG-4 aj.) nebo analogové (například videokazety VHS, Betamax a další.). [1]

Video lze vnímat jako trojdimenzionální pole barevných bodů. Dvě dimenze slouží pro prostorové uložení obrazových informací (vertikální a horizontální) a jedna dimenze reprezentuje časovou doménu. Snímkem rozumíme množinu všech obrazových bodů odpovídajících jednomu časovému momentu. Jedná se tedy o obrázek. [1]

Video obsahuje velké množství redundantních dat a to jak v prostorové, tak i časové doméně, což přímo vybízí ke kompresi videa. Kompresi v prostorové doméně je založena na nedokonalosti lidského oka rozpoznat malé rozdíly v odstínech barvy, lze například využít diskrétní kosinovou transformaci (podobně jako u JPEG komprese). Při kompresi v časové doméně lze vycházet ze vzájemné podobnosti jednotlivých snímků, kdy je často spousta obrazových bodů stejná. [1]

Charakteristickými vlastnostmi jakéhokoliv videa jsou

- počet snímků za vteřinu (*frame rate*)
- rozlišení (*resolution*) – formát HDTV používá až 1920x1080 obrazových bodů
- poměr stran (*aspect ratio*) – typicky 4:3, či např. u HDTV 16:9
- prokládání (*interleaving*) – buď postupné (snímek je vykreslen celý najednou) nebo prokládané, kde videa jsou složena z tzv. půlsnímků, každý snímek je rozdělen na dva půlsnímky trvající polovinu doby celého snímku – první obsahuje liché, druhý pak jen sudé řádky, vykreslování poté probíhá střídavě – lichý/sudý půlsnímek
- datový tok (*bit rate*) – množství přenesených digitálních dat za jednotku času, pro maximalizaci kvality videa při snaze o co nejnižší množství přenesených dat existuje technologie VBR (variabilní datový tok)
- metoda komprese videa (*video compression method*)
- barevný prostor a počet bitů na obrazový bod (*Color space and bits per pixel*)

Soubory, které obsahují video, zvuk, či další datové stopy jako titulky, menu aj., označujeme termínem multimediální kontejnery. Pro hardware či software schopný kódovat/dekódovat datový proud nebo signál používáme termín kodek. [1]

2.1.1 Multimediální kontejner

Kontejner je formát uložení několika proudů multimediálních dat do jednoho souboru (datového toku). V závislosti na vlastnostech multimediálního kontejneru je tedy možno uložit do jednoho souboru i několik vizuálních a zvukových stop, menu, meta-data, obrázky, titulky v různých jazycích aj., s tím, že bude zajištěna synchronizace jednotlivých prvků. Uživatel si posléze v přehrávači zvolí kombinaci multimediálních dat, kterou chce použít. [1]

Kontejnery se navzájem odlišují schopností pracovat s různými multimediálními daty. Aby bylo možné přehrát kontejner, je nutné použít tzv. demuxer (též splitter), který rozdělí datové proudy do různých kodeků a následně do výstupních zařízení. Kontejner sám neříká nic o vnitřní kompresi uložených dat, ta je určena použitým kodekem. Některé kontejnery mohou mít v sobě uloženy pouze omezenou množinu formátů (např. MPEG), jiné jsou tolerantní i k více proudům jednoho typu dat (např. Matroška). [1]

Jelikož kontejnerových formátů existuje celá řada, jsou zde uvedeny pouze ty nejznámější.

2.1.1.1 AVI

Jedná se o zkratku *Audio Video Interleave*, což v překladu znamená *prokládaný zvuk s videem*. Byl vyvinut firmou Microsoft a nejprve se objevil ve Windows verze 3.11 jako součást multimediálního frameworku *Video for Windows*. První videa byla o velikosti pouze 160x120 bodů při 15 snímcích za sekundu, protože tehdejší hardware nebyl tak výkonný, aby zvládl dekompresi v reálném čase. Později došlo k vydání ještě dalších dvou verzí.

Na začátku AVI souboru je hlavička obsahující informace o videu (počet snímků za vteřinu, rozlišení videa, použitá komprese aj.) a o zvuku (vzorkovací frekvence, kodek aj.), na konci se pak nachází tabulka s pořadovým číslem jednotlivých snímků a jejich pozicí v souboru, což má nevýhodu v tom, že soubor nejde přehrát, pokud je neúplný. Tento formát se tedy nehodí pro streaming. Nicméně velkou výhodou AVI je obrovská kompatibilita se všemi OS a rovněž i mobilními zařízeními přehrávajícím video. Velké uplatnění nachází AVI také při editaci videa či jako formát pro ukládání videa digitálními fotoaparáty. [1, 11]

2.1.1.2 MPEG-PS

MPEG-PS je zkratkou pro *MPEG Program Stream*. Od AVI se odlišuje tím, že neobsahuje indexovou tabulku, je vnitřně časově synchronizován. Je používán v případech, kdy je zajištěn bezchybný přenos dat, což je například DVD video, pro které je rovněž využíván jako hlavní formát. Vnitřní struktura tohoto formátu je jednoduchá, čímž zaručuje poměrně snadnou implementaci v komerčních zařízeních. Soubory používající tento kontejner mají obvykle koncovku MPG, VOB nebo EVOB. [1, 13]

2.1.1.3 MPEG-TS

MPEG Transport Stream se na rozdíl od MPEG-PS hodí v prostředích, kde není zaručena bezchybnost přenosu dat, neboť obsahuje mechanismy pro korekci chyb. Nachází tedy své uplatnění pro video streaming. Protože je vnitřně časově synchronizován, je vhodný i pro vysílání na Internetu, kdy lze přehrávání zahájit bez nutnosti mít celý soubor stažený. Soubory jsou obvykle označovány příponami TS, MTS nebo M2TS. [1, 13]

2.1.1.4 Matroška

Matroška pochází z ruského slova *матрешка*, které označuje skládací panenku. Jedná se o multimediální kontejner schopný pracovat s většinou moderních audio a video formátů. Vychází z binární verze jazyka XML (EBML – *Extensible Binary Meta Language*). Díky své struktuře je

schopný nést mnohem více než jen audio/video data. Obsahuje podporu pro titulky, menu, časové značky, fonty písma, kapitoly, lze přiložit i soubory aj. Rovněž nechybí podpora pro uložení více video či audio stop. Velkou výhodou Matrošky je tedy její všestrannost a podpora prakticky jakéhokoliv obsahu. Pro video soubory má příponu MKV, pro audio MKA a pro titulky MKS. Formát je uveřejněn pod licencí GNU LGPL, což znamená, že je volně k použití pro každého. Nejčastější využití Matrošky připadá pro uchování HD videa. [1, 11, 13]

2.1.1.5 Real Media Format

Poměrně starý je i kontejner Real Media Format (RMF) od společnosti RealNetworks. Kontejner je složen z objektů a je vhodný pro streaming. Zajímavým rysem tohoto kontejneru je schopnost pracovat s proměnlivým datovým tokem a v případě nedodání určitého objektu dojde k jeho přeskočení a přehrávání pokračuje na následujícím obdrženém objektu. Typickými příponami souborů jsou RM, RV, RMF pro video a RA pro zvuk. [1, 11]

2.1.1.6 QuickTime

Jedná se o jeden z nejstarších kontejnerů, který byl vyvinut firmou Apple jako konkurence k AVI od firmy Microsoft. Soubory mají příponu MOV. Na svojí dobu to byl poměrně promyšlený formát. Každá stopa obsahuje proud mediálních dat zakódovaných specifickým kodekem anebo se odkazuje na proud dat umístěný v jiném souboru. Datové stopy jsou uspořádány v hierarchických datových strukturách nazývaných atomy. Atom může být rodičem ostatních atomů nebo může obsahovat data, popřípadě je editovat. Nikdy však nemůže mít více funkcí najednou. Každý atom má svoji hlavičku s informacemi o typu a uložení dat. Je velmi blízký kontejneru MP4 (viz 2.1.1.7), který byl na něm založen. [1, 11]

2.1.1.7 MPEG-4 PART 14 (MP4)

Jedná se o multimediální formát kontejneru, který je součástí specifikace MPEG-4. Nejčastěji je nasazen pro ukládání různých audio a video digitálních dat, zejména pak těch, které jsou definovány v MPEG. Na druhou stranu však také umožňuje uložit jiné druhy dat jako titulky nebo obrázky. Rovněž podporuje streaming přes Internet, kdy pro tento účel je v souboru umístěna speciální pomocná stopa. Oficiální příponou je MP4 a z důvodu snadné zapamatovatelnosti je tento kontejner takto často nazýván. MPEG-4 Part 14 vychází z kontejneru QuickTime firmy Apple. Je v podstatě totožný s MOV formátem, ale navíc poskytuje podporu pro *Initial Object descriptors* (IOD) a další MPEG vlastnosti. [1]

Existují různé úpravy tohoto kontejneru, snad jedinou oficiální a značně rozšířenou modifikací je formát 3GPP (3GP, 3GP2) používaný mobilními telefony. Jedná se o do značné míry zjednodušenou verzi navrženou s ohledem na zmenšený úložný prostor a omezenou šířku pásma, tak aby i přenosná zařízení zvládla video soubory přehrát. Typicky se pro obraz používá komprese H.263 a pro zvuk AMR nebo AAC, které se běžně používají pro přenos hovorů. [1]

2.1.1.8 Flash Video

Soubory používají příponu FLV. Jedná se o otevřený formát, který však používá patentované kodeky. Flash video soubory jsou v porovnání s ostatními formáty velmi malé, je použita vyšší míra komprese, nicméně i přesto může být obrazová kvalita vysoká. Pro zvukovou stopu bývá nejčastěji použita MP3 komprese, nechybí však podpora AAC a ADPCM audia. Obrazová data bývají zakódována H.263 kodekem, označovaným jako Sorenson Spark. Od deváté verze Flash přehrávače je podporován formát MPEG 4, který používá H.264 kodek, jenž je vhodný pro video s vysokým rozlišením (*HD video*). [1, 3]

2.1.2 Kodek

Kodekem obecně rozumíme kompresní algoritmus, jehož hlavním cílem je zmenšit velikost datového proudu při snaze o zachování jeho kvality. Dalším požadavkem bývá rychlost provádění komprese/dekomprese, aby bylo možno kódování v reálném čase. Slovo kodek vzniklo ze spojení **kodér+dekodér**. Jedná se buď o hardwarovou nebo softwarovou implementaci určité transformace datového proudu nebo signálu, kterou bývá většinou komprese/dekomprese a to buď ztrátová nebo bezztrátová [1]. Jako příklad hardwarové implementace to může být nějaký čip v přístroji, který zvládne komprimovat i dekomprimovat, příkladem softwarové implementace může být knihovna v počítači, která umožňuje to samé.

Kodek bývá často zaměňován s formátem videa nebo audia. Formát je daný specifikací, která může, ale nemusí být veřejně dostupná. Příkladem formátu specifikace může být například MPEG Layer 3, který je známý pod označením MP3. Kodek je pak konkrétní implementací určitého formátu videa, audia. Kodekem pro MP3 může být třeba LAME, Blade, Xing a další. [1]

Pro základní dělení kodeků se používá druh komprese, kterou je lze třídit na ztrátové a bezztrátové. Dále je možno kodeky klasifikovat například dle velikosti výsledného souboru, datového toku, který může být konstantní nebo proměnlivý.

2.1.2.1 Bezeztrátové kodeky

Při použití bezztrátové komprese zůstávají zachovány všechny informace, výsledkem je tedy naprosto stejná posloupnost bitů jako u originálu. Nicméně jejich použitím nelze dosáhnout vysokého stupně komprese videa, jedná se zhruba o kompresní poměr 2:1 až 5:1 a proto je tedy bezztrátová komprese velmi zřídka používána. Většinou jsou metody bezztrátové komprese postaveny na kódování videa jako série statických snímků, kdy se neuplatňuje predikce. Často slouží jako mezistupeň pro další zpracování videa ztrátovou kompresí, kterou lze dosáhnout přijatelné kvality výstupního videa při mnohem větším kompresním poměru. [14]

Na Huffmanově kódování je postaven HuffYUV kodek s jehož použitím lze v nejlepším případě snížit velikost videa na přibližně 40% jeho původní velikosti. Výhodou tohoto kodeku je pak jeho rychlost. Kompresi barevných modelů RGB a YUV umožňuje dvojice kodeků AVIzlib a AVImszh označovaných pod zkratkou LCL (Loss-less Codec Library) a postavených na algoritmu lz77. Lze dosáhnout většího stupně komprese než při použití kodeku HuffYUV [1].

2.1.2.2 Ztrátové kodeky

Použitím ztrátové komprese dochází ke ztrátě určité informace. Základním přístupem je využití metod způsobujících pro lidské oko neznatelné zkreslení obrazu. Při použití ztrátové komprese se používají tři hlavní směry, kterými lze dosáhnout kompresního poměru 5:1 až 25:1. Jsou to kodeky založené na diskretní kosinové transformaci (např. MPEG1-2, H261-4 aj.), kdy dochází k úpravě obrazu ve frekvenční oblasti. Dále jsou to kodeky používající waveletovou transformaci (Intel Indeo 5 aj.), při které dochází rovněž k modifikacím ve frekvenční oblasti. Poslední skupinu tvoří kodeky, které využívají vektorovou kvantizaci (Sorenson Video aj.) pro seskupování obrazových bodů podobných vlastností. [14]

2.2 Video streaming

Termín streaming je spojen s audiovizuálními daty, kdy dochází k jejich kontinuálnímu přenosu mezi zdrojem a koncovým uživatelem. Streaming může probíhat buď v reálném čase (tzv. živé vysílání – nachází uplatnění u různých televizí či rádií) nebo na požádání (metoda *Video on Demand*) [1]. Následuje přehled nejrozšířenějších řešení vhodných pro video streaming.

2.2.1 Windows Media Services

Windows Media Services představují streaming server pro poskytování videa/audia vysoké kvality na vyžádání či pro doručování datových proudů živého vysílání. Funkce Windows Media Services se mohou odlišovat v závislosti na použité verzi Windows Serveru. Podporovány jsou pouze Windows Media (formáty WMV, WMA, ASF aj.), JPEG a MP3. Kromě streamingu Windows Media Services umožňují datové proudy ukládat do cache paměti či nahrávat. Nechybí podpora různých protokolů, generování statistiky, nastavení řady limitů pro připojení, omezení přístupu, vynucení autentizace, aplikace *Forward Error Correction* (FEC), která zajišťuje kontrolu chyb při datových přenosech, kdy jsou vysílána současně i redundantní data. Příjemce může tedy opravovat chyby bez nutnosti požadavku o opětovné zaslání dat. Běžným přehrávačem je Windows Media Player, který také existuje v podobě zásuvných modulů (pluginů) pro hlavní internetové prohlížeče na platformě Windows. Aktuální verze Windows Media Services 2008 je zdarma dostupná ke stažení pro operační systém Windows Server 2008. [1, 17]

2.2.2 Microsoft SilverLight

Ve spojení se službami Windows Live se jedná o zajímavé řešení pro doručování multimediálního obsahu uživatelům skrze webové stránky. Silverlight streaming je stále ve fázi betatestování a nabízí uživatelům zdarma k vyzkoušení 10 GB prostor. Jednotlivá videa nesmí svou délkou přesáhnout 10 minut a jejich maximální datový tok je stanoven na 1.4Mbps. Měsíční přenos dat pro streaming videa je limitován kvótou 5 TB. Nicméně pro finální verzi se počítá s poskytnutím neomezeného prostoru pro uložení videozáznamů vysoké kvality. Ty budou doplněny o reklamu, pokud uživatel nezaplatí určitý nominální poplatek. Pro funkčnost technologie Microsoft Silverlight je potřeba nainstalovat do systému plugin, který existuje ve verzích pro Windows a Mac. Pro Linux je tento projekt označen názvem Moonlight. Jedná se o svobodnou implementaci frameworku Silverlight vyvíjenou společností Novell. [1, 18]

2.2.3 Helix

Helix je název projektu, jehož cílem je vytvořit software, který zvládne přehrát video či audio mnoha různých formátů a zároveň bude mít schopnost doručovat video/audio přes počítačovou síť. Jedná se o velký open source media framework, který byl inicializován společností RealNetworks. Ta do projektu přispěla velkou porcí kódu a první verze byla vydána v roce 2003. Helix existuje ve verzích pro mnoho operačních systémů a procesorů. Projekt Helix zastřešuje Helix DNA Client a Helix DNA Server.

Helix DNA Client lze chápat jako univerzální stroj, díky němuž lze dosáhnout přehrávání multimédií na prakticky jakémkoliv zařízení. Jedná se pouze o podkladovou vrstvu, která poskytuje funkcionalitu hostujícím aplikacím. Příkladem může být RealPlayer pro Windows, Mac, Linux nebo přehrávač Rhapsody pro Windows.

Helix DNA Server poskytuje streaming a přenos prakticky jakýchkoliv multimediálních typů dat například RealVideo, RealAudio, QuickTime, MPEG-4, MP3 a další s využitím protokolů RTP/RTSP nebo HTTP. [1, 16]

2.2.4 QuickTime Streaming Server

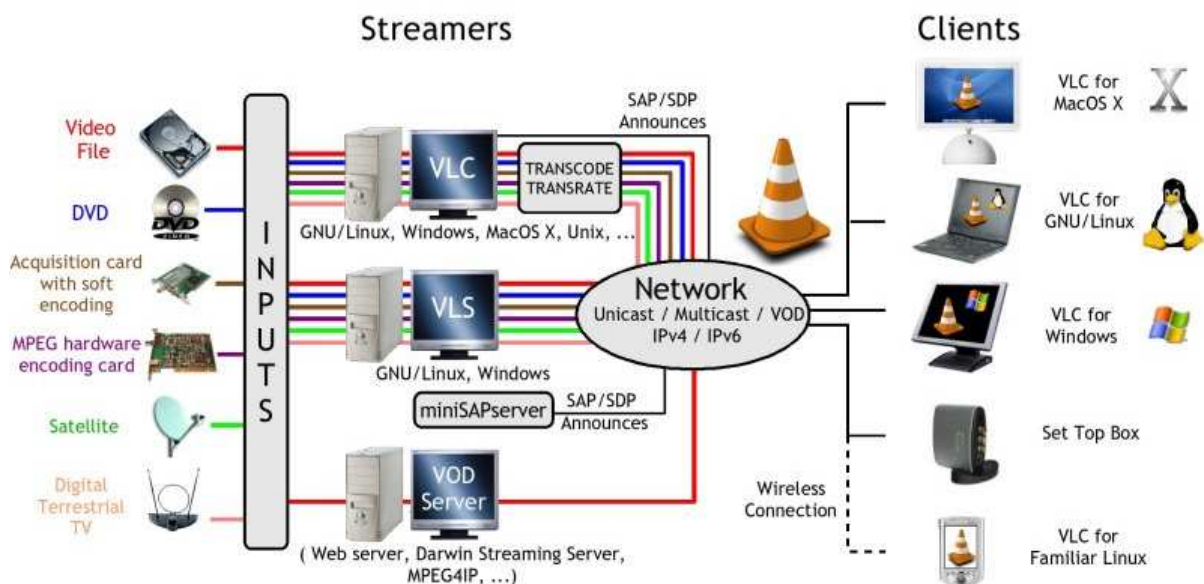
QuickTime Streaming Server (QTSS) je server nebo služba vestavěná do Mac OS X Server, která doručuje video nebo audio uživatelům na požadavek přes počítačovou síť a to včetně Internetu. Ve spojení s QuickTime Broadcaster je schopný vysílat živě. QTSS plně podporuje streamovací

protokoly RTP/RTSP. Open source alternativou ke QTSS je Darwin Streaming Server, který může být zkompileován pro mnoho operačních systémů včetně Windows a Linuxu. [1, 15]

2.2.5 VideoLAN streaming

VideoLAN je kompletní softwarový balík pro video streaming vyvíjený studenty *Ecole Centrale Paris* a vývojáři z celého světa pod licencí GPL (GNU General Public Licence). VideoLan byl navržen pro streaming MPEG videa na sítích s vysokou šířkou pásma. [7]

Tento typ řešení streamingu probíhá přes program VLC media player, který může fungovat buď jako klient nebo jako server. Velkou výhodou programu VLC (původně VideoLAN Client) je možnost streamovat skoro vše, co zvládne přečíst. Druhou možností je použití speciálního streaming serveru VLS (VideoLAN Server), který je schopný streamovat MPEG-1, MPEG-2, MPEG-4 soubory, digitální televizní a satelitní kanály, DVD a živé vysílání na síti s využitím multicastu nebo unicastu. Nicméně většinu funkcionalitu VLS obsahuje taky VLC, a proto je doporučováno použití přímo VLC media playeru. Není potřeba žádného externího programu nebo kodeku, aby program pracoval. [7]



Obrázek 2.1: Řešení streamingu s využitím VideoLAN [převzato z 7]

VideoLAN může být nasazen prakticky na jakoukoliv velikost sítě, například malá síť s využitím jednoho 10/100Mb switchu nebo síť velká jako celý Internet. Nechybí samozřejmě plná podpora IPv6. Velkou výhodou je multiplatformní VLC media player, který existuje ve verzích pro Windows, Linux, Mac OS X, BeOS, Solaris aj. S použitím Mini-SAP serveru je možno přidat k jednotlivým kanálům informace založené na SAP/SDP standardech. Mini-SAP server vysílá oznámení o multicast programech na sítích IPv4 nebo IPv6 a VLC je schopen tyto zprávy přijímat a automaticky přidávat ohlášené programy do playlistu. Mini-SAP server funguje pod Linuxem nebo Mac OS X. [7]

Pro webové prohlížeče je vyvíjen Mozilla plugin, s jehož použitím je umožněno využívat VLC player přímo ve webových stránkách. Základní ovládání přehrávače lze pak provádět přes JavaScript. Rovněž je možno používat ActiveX prvek VLC přehrávače, nicméně toto s sebou přináší určitá omezení. Z důvodu bezpečnosti je instalace ActiveX prvků nedoporučována, a to i když jsou opatřeny digitálním podpisem. Například Microsoft Internet Explorer přestal standardně podporovat instalaci ActiveX prvků. [8]

2.2.6 Flash video streaming

Flash video je souborový formát označovaný jako kontejner (viz 2.1.1.8), který slouží k doručení videa přes Internet s využitím Adobe Flash přehrávače (původně vytvářen společností Macromedia) [1]. Flash zastává velmi významnou roli mezi streaming technologiemi už díky tomu, že má nejvíce podporovaný multiplatformní přehrávač [2]. V současné době je Flash přehrávač dostupný na více než 95 % všech počítačů, které mají přístup k Internetu [4]. Obrovskou výhodou je Flash plugin pro všechny hlavní webové prohlížeče, díky němuž se technologie Flash rychle rozšířila. Flash video je podporováno i programy třetích stran jako VLC přehrávač, Mplayer, QuickTime nebo jakýmkoliv programem využívajícím DirectShow filtry – Windows Media Player a další (musí být přítomný *ffdshow* filtr) [1]. Mezi významné uživatele Flash videa patří Youtube, Google, Yahoo, MySpace, v České republice potom například Stream, N-joy aj.

Flash video bývá často označováno zkratkou FLV. Pro doručení videa na stranu klienta lze v zásadě uplatnit tři metody: vložení videa do SWF souboru, progresivní stahování a streaming.

2.2.6.1 Vložení videa do SWF souboru

Tento způsob se používal zejména v počátcích Flash technologie. Jeho použití se doporučuje pouze v případě, že je potřeba zobrazit video uživatelům, kteří mají nainstalován Flash přehrávač verze 5 a nižší nebo pokud je videosekvence velmi krátká. Celý soubor se nejprve musí stáhnout a pak až může začít přehrávání videa. Pro spuštění videa musí být kompletní obsah souboru uložen do paměti. U videosekvencí delších než 120 sekund může pak dojít k problémům se synchronizací mezi obrazovou a zvukovou částí. Rozhodně není vhodné používat vložené video (*embedded video*) v případech, kdy může být jeho obsah často měněn. Pro uskutečnění jakékoliv změny je nutné opětovně vytvořit a zveřejnit nový SWF soubor. Mezi výhody pak patří použití pouze jednoho souboru a možnost hostování na jakémkoliv běžném webovém serveru. [3]

2.2.6.2 Progresivní stahování

S příchodem Flash přehrávače verze 7 byl uveden nový způsob doručení video obsahu s využitím tzv. progresivního stahování (*progressive downloading/streaming*). Video obsah (FLV, MPEG-4) je uložen odděleně od SWF souboru. Díky tomu lze snadno měnit video bez nutnosti opětovného vytvoření SWF souboru. Přehrávání videa lze zahájit již tehdy, pokud byl stažen a uložen do cache paměti první video segment. Toto chování lze však ovlivnit jazykem ActionScript. Během přehrávání je video načítáno do SWF souboru bez jakýchkoliv omezení týkajících se délky nebo velikosti souboru. Rovněž nedochází k problémům s nedostatkem paměti a se synchronizací mezi obrazem a zvukem. [3]

2.2.6.3 Streaming

Pro streaming je potřeba instalace speciálního softwaru (např. Flash Media Server), který rozšiřuje funkcionalitu Flash formátu. Při jeho použití dochází k ustanovení tzv. perzistentního spojení mezi serverem a koncovým uživatelem. Při běžném server/klient spojení dochází k periodickému zasílání požadavků na server ze strany klienta, kdežto při perzistentním spojení jsou server a klient ve stálém vzájemně kontrolovaném kontaktu.

Toto přináší s sebou výhody v podobě detekce šířky pásma, díky čemuž je možno zvolit odpovídající kvalitu video streamu. Dále je možno detailně monitorovat stav systému a vytvářet tak rozličné statistiky. K přehrávání videa dojde dříve než při jakékoliv jiné metodě, přehrávání může začít kdekoliv bez nutnosti stáhnutí celého souboru. S tím jsou kladeny nižší nároky na klienta v podobě menší potřeby operační paměti a diskového prostoru, neboť na disk není ukládán celý soubor. Jsou lépe využity prostředky sítě, klientovi jsou zasílány jen ty části souboru, které jsou

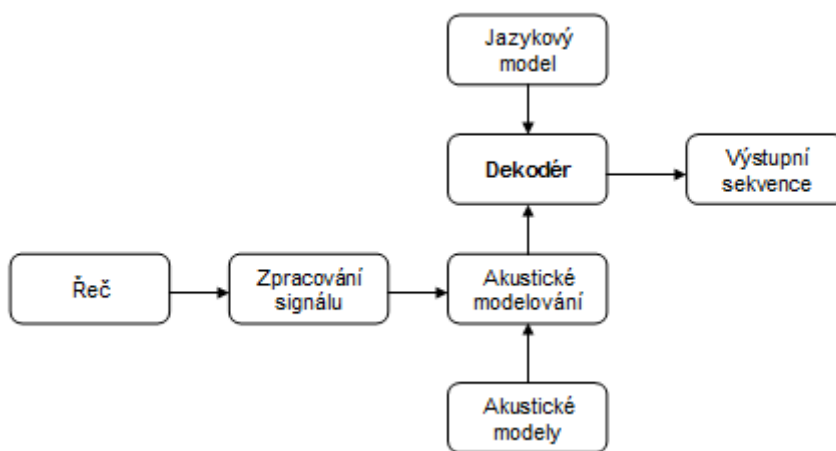
nutné. Je možno vysílat živě, což nachází své uplatnění například při video konferencích, chatech apod. Video soubory při streamingu nejsou ukládány do cache paměti klienta, lze tak chránit jejich obsah před ukradením. Navíc je možno zvýšit úroveň zabezpečení šifrovaným streamingem. K nevýhodám patří cena Adobe Flash Media Serveru. Pro nižší náklady je možno využít hostování u různých poskytovatelů video streamingu. Nicméně dnes existují i nekomerční varianty, které mohou uživateli dostatečně posloužit pro určité potřeby (např. RED5 server). [2]

2.3 Řečové technologie Speech@FIT

Tato podkapitola stručně prezentuje některé technologie výzkumné skupiny Speech@FIT, která sídlí na Fakultě informačních technologií VUT v Brně a zabývá se zpracováním řeči. Jsou zde uvedeny pouze ty, které najdou uplatnění pro prohlížeč přednášek.

2.3.1 Rozpoznávání řeči s velkým slovníkem LVCSR

Jedná se o celkový přepis řeči. Systém dostane určitá audio data a jeho cílem je přepsat vše, co bylo na vstupu. Tento systém lze například použít pro vygenerování přepisu mluveného slova z přednášek. Při rozpoznávání se využívají akustické modely a jazykový model. Akustické modely určují pravděpodobnost, že vstupní signál náleží k akustickým jednotkám (tzv. fonémům). Aby toto fungovalo, je potřeba akustické modely trénovat na velkých řečových databázích. Jazykový model se stará o lingvistickou stránku textu, určuje pravděpodobnost sekvencí slov a trénuje se na velkých korpusech textu. Dekodér poté dává všechny informace dohromady a vyhledává optimální sekvenci slov. Ve výstupu LVCSR se nedá kvůli chybám dobře číst, nicméně lze v něm hledat. [5]



Obrázek 2.2: Schéma systému LVCSR [5]

2.3.2 Detekce klíčových slov nebo frází

Úkolem je detekovat klíčová slova nebo fráze ve zvukových záznamech. Offline detekce spočívá v prohledávání velkých audio archivů, kdy potřebujeme dolovat informace z audia. Online (*real-time*) detekce slouží například pro sledování odposlechů, což nachází uplatnění například u bezpečnostních složek. Hlavním problémem je odmítnat falešné záchyty. Používané přístupy [5]:

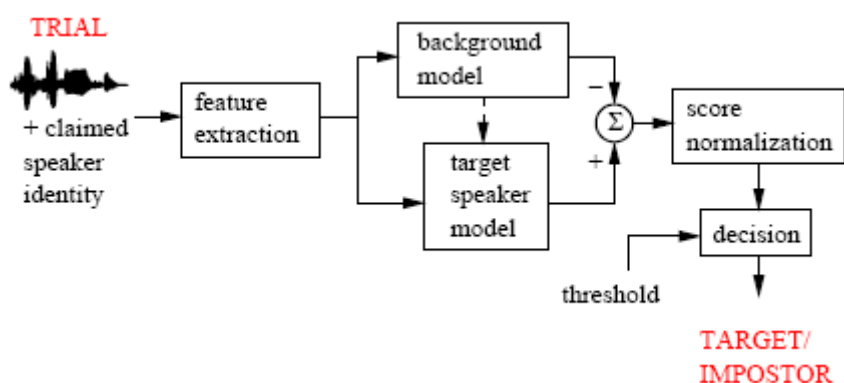
- **Prohledávání výstupu LVCSR** – systém je omezen slovníkem jazyka, LVCSR je docela složité, nicméně vyhledávání je rychlé a je umožněna snadná indexace.
- **Akustický KWS** – jednodušší, modely klíčových slov jsou postaveny v době zadání dotazu, porovnáváním s akustickými daty se získává věrohodnost, další věrohodnost se odebrává z modelu pozadí. Obě věrohodnosti se srovnávají a výsledkem je křivka,

po jejímž prahování lze zjistit, kde klíčová slova byla ve zvukovém záznamu vyslovena. Tento přístup řeší problém omezené slovní zásoby, lze zpracovávat i cizí slova. Jelikož je systém relativně jednoduchý, může běžet online. Problémem je však indexace, protože před začátkem vyhledávání systém nezná slova, která se budou hledat. Tento přístup je nevhodný pro prohledávání velkých archivů a nemá sílu jazykového modelu, dochází k častým chybám u krátkých slov.

- **Vyhledávání na fonémových grafech** – podobným rozpoznávačem jako u LVCSR se rozpoznává místo slovního grafu fonémový graf (fonémová latice) a vyhledává se sekvence fonémů. Tento přístup řeší problémy s omezenou slovní zásobou a poskytuje možnost indexace fonémů. Na druhou stranu je takovýto systém složitější a pomalejší než při použití přístupu založeného na LVCSR.

2.3.3 Identifikace a verifikace mluvího

Při identifikaci mluvího se porovnávají parametry hlasu s databází řečových vzorků ve snaze prokázat podobnost řečového signálu. Výsledkem je přiřazení určitého zvukového segmentu buď jedné osobě z množiny mluvích, nebo osobě žádné. Verifikaci mluvího ověřujeme, zdali parametry hlasu mluvího odpovídají parametrům hlasu již dříve analyzovaného [12]. Jde tedy o ověření identity osoby (*Je to opravdu pan Novák?*). [5]



Obrázek 2.3: Schéma verifikace mluvího [převzato z 5]

V systému se používá extrakce parametrů (*features extraction*), což je množina čísel určitým způsobem parametricky popisující řečový signál [12]. Následně jsou tyto parametry porovnávány s modely, které jsou natrénovány na množině mluvích. Modely poskytují určitou věrohodnost, odečtou se a získané skóre se normalizuje a je porovnáno s detekčním prahem, který určí, zdali se skutečně jedná o onu osobu.

2.3.4 Rozpoznávání jazyka

Cílem rozpoznávání jazyka (*language identification*) je určení jazyka, kterým bylo v určitém zvukovém záznamu mluveno. Tato technologie nabízí široké uplatnění například v různých bezpečnostních složkách, telefonních centrech, kdy je potřeba spojit osobu na lince s operátorem, který hovoří stejným jazykem. Dalším příkladem použití může být automatická aplikace příslušného rozpoznávače řeči, který se s jazykem různí.

2.3.5 Indexování a vyhledávání v řečových datech

Zvukový záznam je převeden rozpoznávačem řeči do textové formy, a následně je v těchto datech vyhledáváno. Za nejjednodušší systém pro vyhledávání v řečových datech můžeme považovat takový,

který je postaven na obyčejném vyhledávání v textu. Pokud chceme přidat nová data do systému, je potřeba zpracovat zvukový záznam rozpoznávačem řeči. Tím obdržíme na výstupu textový přepis řeči. Následně lze na tyto data aplikovat jakýkoli systém pro textové vyhledávání. Tento typ vyhledávače však může fungovat dobře jen tehdy, pokud je textový přepis bezchybný, což je pro počítačové zpracování řeči těžko řešitelný problém. Nicméně dnes se takovýto systém v praxi často používá. Pro vyhledávání ve kvalitních zvukových záznamech, kdy předpokládáme nízkou četnost chyb (například televizní zprávy, předpověď počasí, oznámení aj.), může být však jeho nasazení dostatečné. [6]

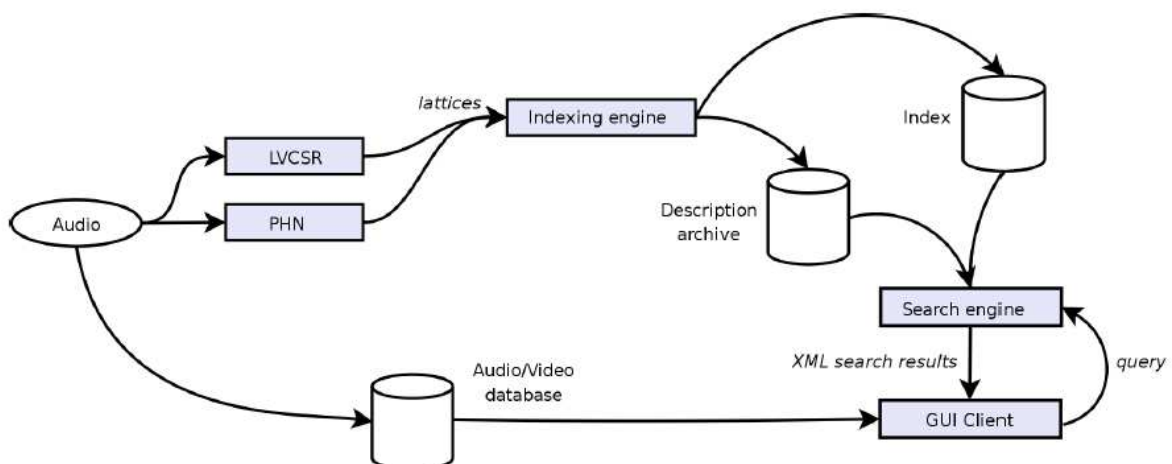
2.3.5.1 Rozdělení systémů pro vyhledávání v řeči

Speciální vyhledávací systémy v řečových datech lze rozdělit do několika kategorií podle přístupů, na kterých jsou založeny [6]:

- **LVCSR** (Large Vocabulary Continuous Speech Recognition) – vyhledávací systém indexuje výstup LVCSR systému. Nevýhodou je omezená slovní zásoba. Rozpoznávač není schopen zpracovat slova, která nejsou ve slovníku. Při vyhledávání tedy tato slova nemohou být nalezena, což vede k nižší schopnosti získání dat.
- **Rozpoznávač fonémů** (phoneme recognizer) – indexuje výstup slovních jednotek získaných fonémových rozpoznávačem, který na rozdíl od systémů založených na LVCSR nepracuje se slovníkem a tedy může rozpoznat i méně častá slova, či slova, která v daném jazyku vůbec neexistují. Na druhou stranu systémy založené na fonémových rozpoznávačích jsou méně přesné. Hledaný řetězec (dotaz) musí být převeden na fonémy, aby bylo možno vyhledávat v datech.
- **Kombinace LVCSR a rozpoznávače fonémů** – tento systém využívá výhod dříve uvedených systémů – přesnosti u systémů založených na LVCSR a robustnosti u systémů využívajících fonémový rozpoznávač.

2.3.5.2 Vyhledávací stroj skupiny Speech@FIT

Systém vyvíjený skupinou Speech@FIT využívá pro zpracování audia kombinaci přístupů založených na LVCSR a fonémovém rozpoznávači. Výstupní data jsou naindexována a uložena do databází. Soubory s indexy jsou uloženy do *Index* databáze a lattice (orientovaný graf) do *Description archive* databáze. V systému je ještě multimediální databáze, ze které je možno načítat audio/video data.



Obrázek 2.4: Návrh vyhledávacího stroje v řeči [převzato z 6]

Vyhledávací stroj může fungovat buď jako samostatná aplikace nebo neblokující server. V server módu klient pošle dotaz na adresu serveru (specifikována IP adresou a portem) a ten mu odpoví výsledky uloženými v XML formátu. [6]

2.4 Zpracování slajdů z videozáznamů přednášek

V této podkapitole je popsán postup pro automatické zpracování slajdů z videozáznamů přednášek a značná část těchto informací byla převzata z [27]. Tento proces spočívá v nalezení oblasti promítaného slajdu ve videozáznamu a následné aplikaci vhodných obrazových transformací, kterými lze postupně extrahovat jednotlivé slajdy z videa s ohledem na zamezení jejich případné duplicity. Základním požadavkem je kvalitní videozáznam pořízený při dobrých světelných podmínkách.

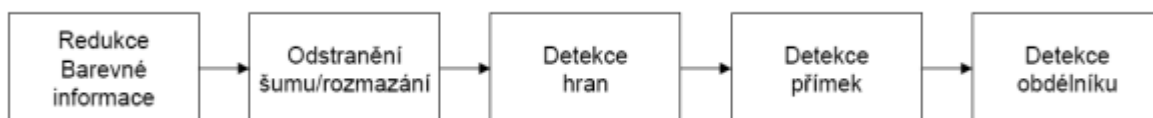
Extrakcí slajdů z videozáznamu lze získat informace, kdy byl který slajd promítán. Tyto údaje lze využít pro vytvoření časové osy s příslušejícími slajdy. Pomocí porovnávání vzájemné podobnosti obrázků lze nalézt odpovídající slajdy v souboru s prezentací a synchronizovat je s videozáznamem. Dále automatická extrakce slajdů z videozáznamů najde své uplatnění v případech, kdy k přednášce není k dispozici původní soubor s prezentací. Z extrahovaných slajdů tak uživatel získá možnost se jednoduše přenést na místo v záznamu, kde byl poskytnut výklad k danému slajdu. Rovněž využití připadá pro indexování záznamů přednášek.

2.4.1 Návrh základního systému

V prvé řadě je potřeba uvažovat pouze takové videozáznamy, které jsou pořízeny statickou kamerou. Tedy pozice plátna s promítanými slajdy je neměnná. Dalším předpokladem je výrazný rozdíl v jasu mezi hranicemi plátna a okolím. Lze předpokládat, že plátno je obdélníkového tvaru, nicméně umístěním kamery je potřeba brát v úvahu možné zkreslení způsobené perspektivní projekcí. Tím se poté promítací plátno ve videozáznamu jeví jako lichoběžník.

2.4.1.1 Vyhledání plátna

Na vstupu je několik náhodných snímků získaných z různých míst videozáznamu. Na každý tento snímek se aplikuje sekvence grafických algoritmů schématicky znázorněných na následujícím obrázku 2.5. Cílem těchto postupů je v obraze detekovat přímky a následně dle těch nejdelších, které tvoří obdélníkový tvar, stanovit souřadnice plátna. Dále se porovnají souřadnice s ostatními snímky a určí se výsledná pozice plátna.



Obrázek 2.5: Schéma postupu pro detekci pozice plátna ve videozáznamu [převzato z 27]

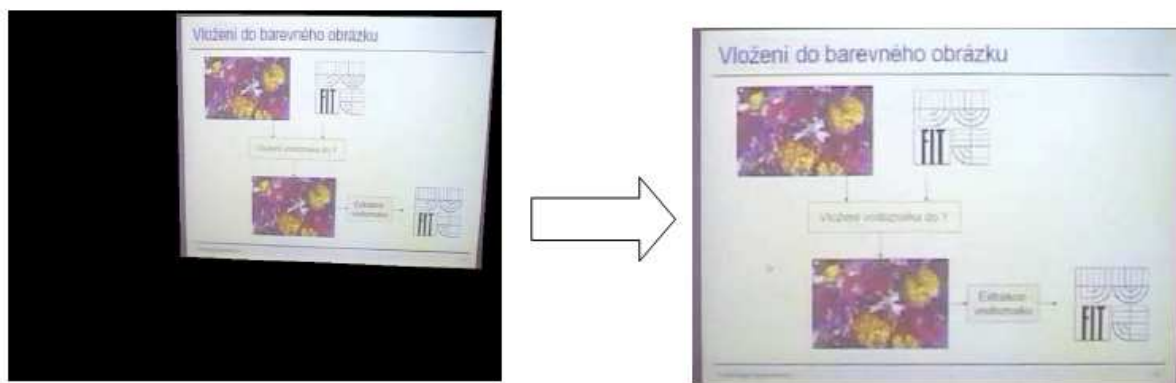
2.4.1.2 Korekce perspektivní projekce

Jelikož neznáme přesnou pozici kamery, projektoru a plátna, je potřeba najít postup pro provedení perspektivní korekce, která částečně odstraní zkreslení obrazu. Obraz je deformován dvěma transformacemi, kdy souřadnice jakéhokoliv bodu ze slajdu (x,y) jsou při promítání projektorem transformovány na neznámou pozici na plátně. A následně je každý bod z plátna transformován na odpovídající pozici (X,Y) do obrazu snímáného kamerou. Cílem této korekce je tedy nalezení vhodného mapování souřadnic z (x,y) na (X,Y) , jinými slovy je potřeba najít transformaci přímo z oblasti slajdů do oblasti obrazu v kameře.

Na první pohled se toto mapování může díky mnoha neznámým jevit jako neproveditelné. Naštěstí můžeme využít skutečnosti, že všechny body jsou promítány na společnou rovinu, čímž můžeme stanovit homografii mezi kamerou a projektorem. Složenou transformaci mapování (x, y) na (X, Y) lze zapsat v homogenních souřadnicích jako:

$$\begin{pmatrix} xw \\ yw \\ w \end{pmatrix} = \begin{pmatrix} p_1 & p_2 & p_3 \\ p_4 & p_5 & p_6 \\ p_7 & p_8 & p_9 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}, \text{ kde } \vec{p} = (p_1 \dots p_9)^T, |\vec{p}| = 1 \quad (2.1)$$

Hodnoty \vec{p} lze určit již ze 4 odpovídajících bodů, pokud ovšem žádné tři body nejsou kolineární¹ [28]. Pro tento účel lze například použít známé body získané detekcí pozice plátna ve videozáznamu.



Obrázek 2.6: Výsledek aplikace perspektivní korekce [převzato z 27]

2.4.1.3 Detekce změny slajdů

Cílem je zachytit stav, kdy dojde ke změně slajdu. S nastavenou přesností je po každých x snímcích extrahován obsah plátna a ten je porovnáván s předchozími stavy. Výchozím předpokladem je, že na plátně jsou promítány statické snímky, tedy takové, jejichž obsah je neměnný. Při jejich přepnutí zároveň systém očekává výraznou změnu. Pokud výsledkem porovnání je koeficient přesahující určitý práh, je tento stav označen jako změna slajdu. Pro vzájemné porovnávání snímků lze využít různých statistických hodnot (minimum, maximum, suma čtverců, střední hodnota aj.), které jsou posléze prahovány určitým konstantním/dynamickým prahem. Problémem jsou různé rušivé stavy, například když před plátnem objeví postava přednášejícího apod.

2.4.1.4 Výstup systému

Jako základní výstup takovéto aplikace lze považovat osu přednášky s body, ve kterých došlo ke změně obsahu plátna. Pro jednotlivé body lze extrahovaný snímek uložit do souboru. Rozšířením funkcionality aplikace lze extrahované snímky poté přiřadit pomocí metod pro vzájemnou podobnost obrázků k jednotlivým slajdům nacházejících se v podkladech pro přednášky. Těmi nejčastěji bývají PDF soubory. Pro tento účel je však potřeba nejdříve jednotlivé slajdy umístěné uvnitř PDF souboru převést do obrázků nějakého rozšířeného grafického formátu, tím pro tuto aplikaci může být například PNG.

¹ Tři nebo více bodů, které leží na téže přímce, se nazývají **kolineární**.

2.4.2 Implementovaný systém

Aplikace, která se stará o automatickou synchronizaci slajdů s videozáznamem, je z velké části postavena na informacích prezentovaných v předchozí podkapitole (viz 2.4.1). Za implementaci této aplikace patří velký dík Ing. Vítězslavu Beranovi.

Na vstupu je potřeba aplikaci v parametrech předat soubor s videozáznamem přednášky, cestu k adresáři, do kterého budou extrahované slajdy ukládány a pak, pro nalezení odpovídajících obrázků k extrahovaným slajdům, také XML soubor pro komponentu slajdy (formát tohoto souboru je popsán dále v 4.6.4). Příklad spuštění aplikace:

```
det_slajds.exe -i ISS_2008-10-03.avi -p slides/detected/ -s slides.xml
```

Průběh zpracování je možné ovlivnit řadou parametrů, které se načítají z příslušného XML souboru. Jedná se například o nastavení kroku definujícího kolikátý každý snímek videozáznamu bude zpracován nebo o rozměry extrahovaných snímků, hodnoty prahů aj.

Výstupem je potom XML soubor obsahující odkazy na extrahované slajdy a pak také na externí obrázky slajdů, pokud k nim byly přiřazeny. Součástí je taky parametr *confidence*, který udává, jak hodně je si program přiřazením jistý. Dále je uveden počáteční a poslední snímek (parametry *start* a *end*). Tyto údaje tedy udávají dobu, po kterou je daný slajd ve videozáznamu zobrazen.

2.4.3 Alternativní řešení

Kromě postupu, který je přímo založen na zpracování slajdů z videozáznamu, lze uvažovat také řešení, které bude získávat potřebná data již v době přednášky. Například se tedy může jednat o určitou utilitu napojenou přímo na prezentační software. V momentě změny slajdu dojde k zaznamenání času a čísla slajdu. Problémem by však byla následná synchronizace se zpracovaným videem, kdy by bylo nutné seřadit časy.

3 Prohlížeče přednášek

Tato kapitola seznamuje čtenáře s podobně orientovanými multimediálními prohlížeči. Jejím úkolem je poskytnout náhled do již existujících řešení a posloužit jako zdroj inspirace pro vytvářený webový prohlížeč přednášek.

3.1 MBrowser

MBrowser představuje zkratku vzniklou ze spojení *Multi-modal Browser* či *Multimedia Browser*. Jedná se o prohlížeč, který je vyvíjen na Fakultě informačních technologií VUT v Brně v rámci projektu *Presentation As Synchronized Experience*. Myšlenkou pro vznik tohoto projektu bylo poměrně velké množství videozáznamů ve vysoké kvalitě, jež jsou pořizovány z přednášek a poskytují zajímavý testovací materiál pro různé rozpoznávače, díky kterým je například umožněno vyhledávat v řečových datech. Právě tato vlastnost může studentům výrazně usnadnit přístup k požadovaným informacím, které je mnohdy problém ve videozáznamech rychle nalézt. [19, 24]

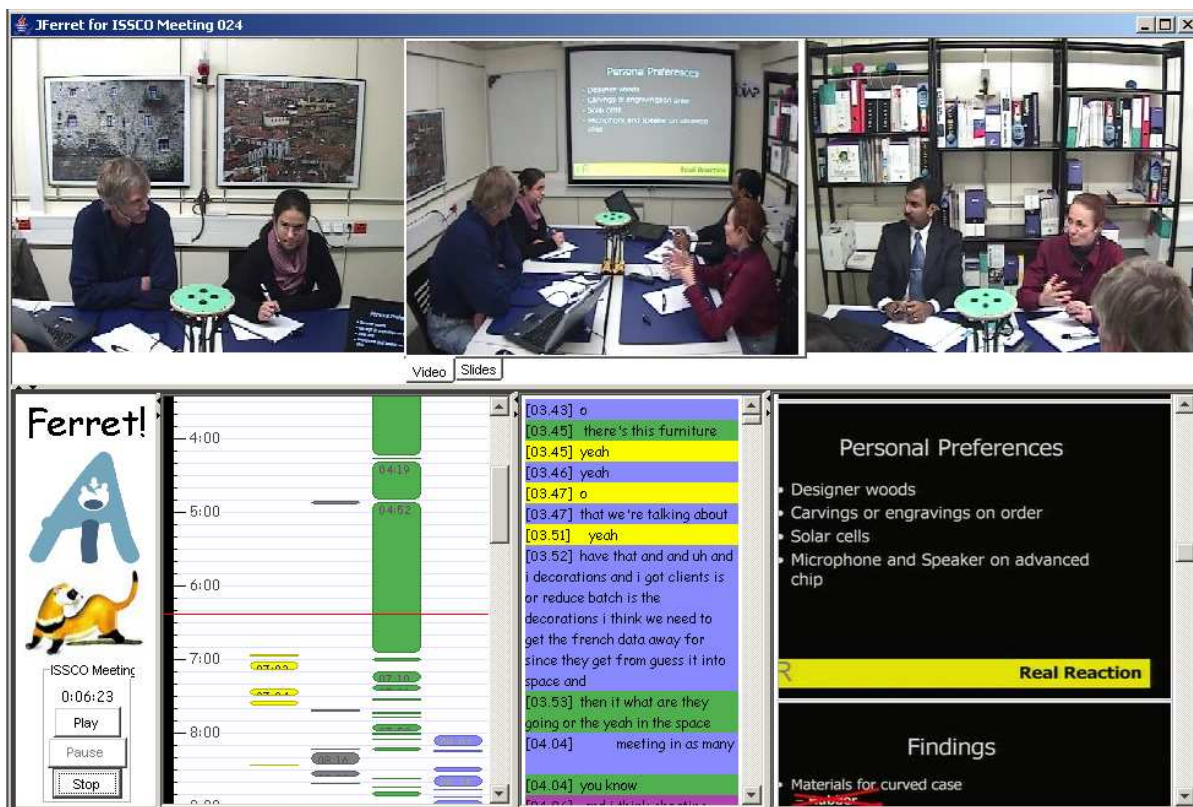
Prohlížeč je vyvíjen s cílem snadné konfigurace a rozšiřitelnosti o nové funkce a technologie. Hlavním datovým formátem je XML, které zde slouží například pro uložení konfigurace jednotlivých komponent, textového přepisu řeči, potřebných dat pro načítání slajdů synchronizovaných s videozáznamem aj. XML je rovněž použito při vzájemné komunikaci komponent, které tento formát používají pro zasílání zpráv. MBrowser je vyvíjen v programovacím jazyku C++ s využitím multiplatformní knihovny wxWidgets, která dodává potřebnou funkcionalitu pro vytvoření grafického uživatelského rozhraní aplikace. Jádro prohlížeče MBrowser poskytuje aplikaci jednoduché komponentové rozhraní. V rámci spuštěné aplikace je možné komponenty dynamicky do prohlížeče přidávat či z něj odebírat. [19, 24]

Tento prohlížeč je stále ve fázi vývoje a měl by využívat stejný indexační a vyhledávací systém v řečových datech, na kterém bude rovněž postaven webový prohlížeč přednášek. Neměla by také chybět vzájemná podpora mezi některými XML datovými soubory a to alespoň pro načítání dat textového přepisu mluveného slova a informací o slajdech.

3.2 JFerret

JFerret je multimediální prohlížeč vyvíjený v rámci AMI projektu *MultiModal Media Fileserver* pod záštitou švýcarského výzkumného institutu IDIAP. Výhodou tohoto prohlížeče je jeho flexibilita, umožňující vytvořit prakticky jakékoliv uživatelské rozhraní s různou kombinací doplňkových pluginů. Konfigurace prohlížeče je načítána z XML souboru, jenž definuje, které komponenty bude prohlížeč používat, jak budou v prohlížeči vizuálně uspořádány a jak budou vzájemně mezi sebou komunikovat. [22]

JFerret je napsán v jazyce Java, což zaručuje přenositelnost mezi platformami, problémem však může být menší aplikační výkon. Výhodou je pak možnost volby, zdali JFerret poběží jako samostatná aplikace, či jako *applet* uvnitř webového prohlížeče. Standardní balík prohlížeče JFerret obsahuje nejrozličnější pluginy pro video, audio, slajdy, anotační časovou osu, transkript mluveného slova aj. Nechybí možnost přidávat do prohlížeče vlastní pluginy, kdy je pro tento účel poskytnuto jednoduché API rozhraní, které také poskytuje elegantní mechanismy pro vzájemnou komunikaci mezi všemi pluginy prohlížeče. [22]



Obrázek 3.1: Ukázka prohlížeče JFerret [převzato z 22]

Tento prohlížeč je sestaven celkem z 29 pluginů (např. 3 pluginy pro video, audio přehrávač a další).

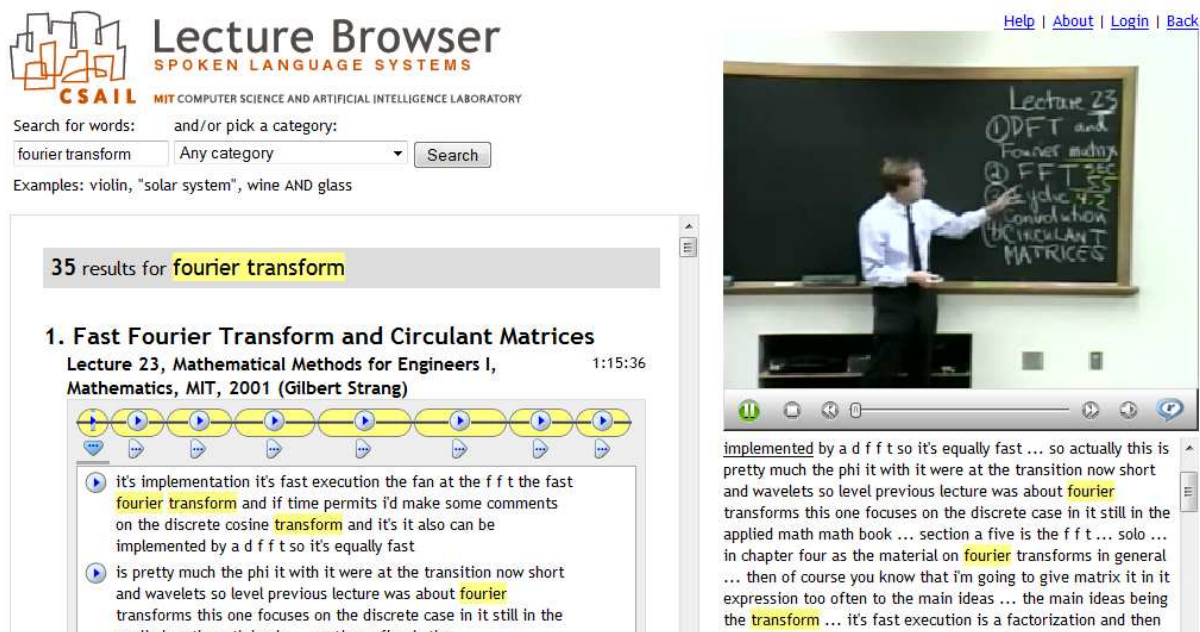
3.3 MIT Lecture Browser

MIT Lecture Browser¹ je produktem *Computer Science and Artificial Intelligence Laboratory* (CSAIL) na Massachusetts Institute of Technology (MIT). Tento prohlížeč je postavený na transkriptech generovaných softwarem pro zpracování řeči. Častým problémem je, že přednášky obsahují mnoho technických pojmů, které se mnohdy nenachází v počítačových slovnících. Proto výzkumníci použili knihy, poznámky z přednášek, abstrakty a další pro nalezení takovýchto slov, které následně přidali do slovníku. K přednostem tohoto prohlížeče patří jeho rozhraní postavené na webových stránkách, které jsou sice jednoduché, nicméně přehledné a dobře splňující svůj účel pro vyhledávání ve videozáznamech. Prohlížeč je dostupný veřejnosti s mnoha záznamy přednášek pořizovaných na MIT. Pro přehrávání videozáznamů je však použit plugin přehrávače RealPlayer pro webovou prohlížeč, což z hlediska jeho rozšíření a podpory není zrovna ideální řešení.

Hnacím motorem prohlížeče je vyhledávací systém v transkriptech přednášek. Vyhledávání lze omezit na určitou kategorii. Výsledkem dotazu je seznam přednášek, na kterých hledané slovo zaznělo. K tomu je zobrazena část textového přepisu obsahující vyhledávaný termín, který je doplněn o odkazy na odpovídající úseky ve videozáznamu. Úspěšnost pro vyhledání požadovaného termínu závisí na kvalitě vygenerovaného přepisu řeči. Pro nerodilé mluvčí klesá úspěšnost rozpoznání k 50 %, pokud systém nebyl natrénován na jeho akcent a slovní zásobu. Nicméně i transkript obsahující velký počet chyb v mezích od 30 % do 50 % může stále dobře posloužit pro vyhledání klíčového slova, ačkoliv pro čtenáře je prakticky nepoužitelný [20]. V módu přehrávání videa je

¹ <http://web.sls.csail.mit.edu/lectures/>

souběžně zobrazován transkript, v němž je zvýrazňováno každé jednotlivé slovo, které je právě vyslovováno.



Obrázek 3.2: Ukázka prohlížeče MIT Lecture Browser [21]

3.4 Videlectures.net

Za zmínku stojí určitě uvést portál Videlectures.net, který volně nabízí k přehrání obrovské množství videozáznamů pořizovaných na různých konferencích, letních školách, seminářích a to mnohdy také v kombinaci se slajdy, které jsou plně synchronizovány s videem. To je nejčastěji k dispozici k přehrání skrze přehrávač Flash videa JW FLV Media Player, případně bývá i často podpora přehrávání videa s využitím webového pluginu programu Windows Media Player. Přednášky jsou přehledně organizovány do různých kategorií.

Cílem tohoto portálu je propagovat vědu, vyměňovat nápady a pomáhat sdílet znalosti. Pro uživatele portálu je nabídnuta možnost hodnotit jednotlivé přednášky, přidávat k nim komentáře. Vznik tohoto prohlížeče je datován do počátků roku 2002, kdy jeho vývoj začal na Josef Stefan Institute ve Slovinsku. Tento portál se stává hlavním úložištěm pro videozáznamy z akademického a výzkumného prostředí po celém světě. [23]

4 Realizace prohlížeče přednášek

Tato kapitola se zabývá realizací webového prohlížeče přednášek. Na počátku stál zajímavý nápad vytvořit portál pro přednášky postavený na technologiích pro zpracování řeči, které vyvíjí výzkumná skupina Speech@FIT. Zejména pak možnost hledání v audio záznamech by měla studentům značně usnadnit nalezení požadovaných informací. Jako základ inspirace pro tento projekt měla posloužit aplikace MBrowser (viz 3.1).

V průběhu práce na webovém prohlížeči přednášek bylo potřeba se přenést přes mnoho překážek. Ať již se to týká implementačních rozdílů v rámci internetových prohlížečů, problémů technického rázu aj. Rovněž díky pomoci a nápadům dalších osob však vznikl použitelný produkt, který si zaslouží alespoň pozornost.

4.1 Počáteční fáze

Dle pokynů v zadání diplomové práce bylo nejprve nutné se seznámit s multimediálním přehrávačem MBrowser (viz 3.1). Konkrétně se jednalo o nastudování formátu datových souborů a metod vzájemné komunikace jednotlivých komponent. Byla zde totiž myšlenka, aby zejména XML soubory, se kterými pracuje MBrowser, byly použitelné i pro webový prohlížeč přednášek. Tedy měly vzniknout dvě verze prohlížečů přednášek. Ta první by vycházela z řešení současného přehrávače MBrowser. Jednalo by se o klasickou aplikaci napsanou v jazyce C++. Zatímco druhá verze by měla být postavena na webových stránkách, a tím široce dostupná přes hlavní internetové prohlížeče na různých platformách. Tato druhá verze je předmětem této práce.

Velice podstatnou otázkou bylo vyřešení streamingu videa. Bylo nutné nastudovat a porovnat nejrůznější řešení použitelné pro webové rozhraní. Bylo otestováno několik možností doručování videozáznamu uživateli. Jednalo se například o řešení přes VLC Mozilla plugin, který byl schopen přehrávat video ze současných video serverů FITu, na kterých je spuštěn RTSP (Real-Time Streaming Protocol). V tomto případě by nebyla potřeba žádného speciálního streaming serveru. Dalším způsob představovalo řešení skrze Real Media Player, nicméně podobně jako u VLC pluginu se vyskytly problémy s funkčností v jiných internetových prohlížečích. Ovládání přes JavaScript fungovalo pouze na operačním systému Windows a ne všechny internetové prohlížeče byly podporovány. Následovalo zaměření na technologii Flash, která je v tomto ohledu hráčem číslo jedna. Nejprve bylo potřeba najít způsob, který by zdarma poskytoval streaming Flash videa. Adobe Flash Media server je placený. Po pátrání byl objeven RED5 server, který podporuje streaming Flash videa a je zdarma. Díky však jeho značné složitosti a několika problémům bylo nakonec rozhodnuto vyzkoušet variantu Lighttpd serveru ve spojení se speciálním zásuvným modulem *mod_flv_streaming* (viz 4.10.2.1), který umožňuje pseudo-streaming. Toto řešení se ukázalo jako ideální pro potřeby tohoto projektu.

Následoval vývoj prvotní verze prohlížeče. Ta poskytovala uživateli jednoduchou webovou stránku, na které byl zobrazen transkript synchronizovaný s videozáznamem. Vybráním určitého textového segmentu transkriptu došlo v přehrávači k přenesení na patřičné místo ve videozáznamu. Dokončením prvotní verze webového prohlížeče přednášek byly sestaveny požadavky a stanoveny cíle pro výsledný systém této diplomové práce. Ty byly s následujícím vývojem dále upravovány a rozšiřovány.

4.2 Cíle projektu

Hlavním cílem je vytvoření multifunkčního prohlížeče s důrazem na poskytnutí informací k dané problematice probírané na přednášce. Zároveň tyto informace musí být snadno přístupné. Příkladem může být přednáška, která obsahuje videozáznam, k němu odpovídající slajdy a doplňující informace například v podobě odkazů na vhodnou literaturu, články na webu, fórum diskutujících studentů o dané problematice apod.

Na audio stopu videozáznamu budou nasazeny nejruznější technologie pro zpracování řeči. Výstupem bude přepis mluveného slova s rozlišením jednotlivých mluvčích. Bude ustanoven způsob komunikace mezi prohlížečem a serverem pro vyhledávání v audio datech. Důležitým aspektem prohlížeče má být orientace na multimédia a proto budou jednotlivé typy dat rozříděny dle svého zaměření do komponent. Ty bude uživatel mít možnost zobrazit/schovat dle své potřeby. Videozáznam přednášky bude synchronizován se slajdy a s transkriptem. Budou vytvořeny dvě jazykové mutace prohlížeče – česká a anglická, tím bude systém více uživatelsky přívětivý a dostupný i mezinárodním studentům. Projekt bude využívat řešení, která jsou zdarma dostupná. To se týká například programů potřebných pro úpravy videozáznamů, slajdů nebo třeba hostování celého systému aj.

Jelikož se jedná o webový prohlížeč přednášek, tak vytvořený portál musí bez problémů podobně fungovat v dnes nejvíce používaných internetových prohlížečích, mezi které nepochybně patří IE, Firefox a Opera. Stránky budou sestavovány dynamicky, jejich kód bude odpovídat standardu XHTML. Potřebná data budou načítána z XML souborů. Prohlížeč musí být jednoduše rozšiřitelný o další komponenty a funkcionalitu. Jako ukázková množina dat bude sloužit přednáškový kurz předmětu Signály a systémy. K pořízeným videozáznamům budou rovněž přiřazeny slajdy, které budou náležitě zpracovány. Výsledný produkt musí být otestován zejména z hlediska bezpečnosti a použitelnosti.

Z výše definovaných cílů a požadavků byl navržen a implementován prohlížeč přednášek. Následující část textu pojednává o vytvořeném systému, rozebírá jeho architekturu, technické řešení, uživatelské rozhraní, komponenty, datové soubory a další věci týkající se implementovaného prohlížeče přednášek.

4.3 Architektura systému

Z hlediska architektury systému jsou soubory v prohlížeči přednášek rozříděny do jednotlivých adresářů dle svého zaměření.

- **css** – obsahuje soubory se stylpisem stránek, *layout.css* – definuje stylpis pro celý webový prohlížeč přednášek kromě jednoho případu. Tím je speciální stránka, která zobrazuje všechny obrázky slajdů ve speciálním okně, pro ni je stylpis načítán ze souboru *slides-layout.css*.
- **data** – tento adresář slouží pro datovou část prohlížeče přednášek. Ústředním souborem definujícím, které přednášky prohlížeč obsahuje, je *lecture_list.xml* (viz 4.5.2). V jeho podadresářích jsou uložena data pro jednotlivé přednášky.
- **images** – slouží pro umístění obrázků. V podadresáři *layout* se nachází obrázky, na kterých je postaven vzhled stránky.
- **javascript** – nachází se v něm soubory pro JavaScript. Soubor *swfobject.js* slouží pro detekci Flash přehrávače v prohlížeči a nachází uplatnění pro vnoření přehrávače JW FLV Media Player (viz 4.10.2.2). Dalším souborem je *main.js*, který obsahuje zbytek kódu jazyka JavaScript použitého v prohlížeči přednášek.

- **lang** – slouží pro umístění XML souborů s jazykovou lokalizací stránek (viz 4.5.1)
- **logs** – je vytvořen pro umístění logů prohlížeče přednášek. Jedná se o uložení dat použitých pro vyhledávací server v audio datech. Analýzou dotazů zasílaných na server lze dále vylepšovat funkcionalitu systému.
- **mediaplayer** – v tomto adresáři je umístěn JW FLV Media Player společně s dalšími soubory, které tento přehrávač využívá. Konkrétně se jedná o zásuvný modul pro titulky a soubor definující vzhled přehrávače.
- **php** – obsahuje PHP soubory, ze kterých jsou dynamicky sestavovány webové stránky prohlížeče přednášek. V podadresáři *classes* jsou soubory, ve kterých je použit objektově orientovaný PHP kód. Pokud název souboru končí před svoji příponou řetězcem *Page*, jedná se o kód použitý pro sestavování XHTML výstupu. Tedy jde o soubory, které poskytují prezentační vrstvu systému. Ostatní soubory obstarávají zejména logickou funkcionalitu, načítání/ukládání dat, komunikaci aj. Významným souborem je *Settings.php*, který obsahuje nastavení významných hodnot ovlivňujících chod prohlížeče přednášek. V podadresáři *utilities* se nachází soubory, které slouží pro přípravu některých dat pro komponenty Transkript, Slajdy a Časová osa přednášky.

V kořenovém adresáři se nachází PHP soubory, které sestavují daný typ webových stránek (např. stránka zobrazující výpis přednášek v dané kategorii, stránka pro přehrávání dat přednášky aj.). Hlavním souborem, který mimo jiné slouží jako domovská stránka a také jako rozcestník pro přístup k dalším stránkám, je *index.php*.

4.4 Kategorizace obsahu

Každá přednáška je přiřazena do určité kategorie. Třídění přednášek je nezbytné z důvodů poskytnutí snadné orientace v prohlížeči, usnadnění vyhledávání a seskupení podobných přednášek. Základní kategorií v prohlížeči přednášek je školní rok a příslušný semestr, pod který daná přednáška spadá. Neméně důležitou kategorií je pak předmět, pro který byla přednáška pořízena. K dalším údajům, dle kterých lze obsah třídit, patří datum pořízení přednášky, jméno přednášejícího, probírané téma aj.

Systém je založen na hierarchické kategorizaci obsahu. Pro snazší navigaci je v prohlížeči zobrazeno tzv. drobečkové menu informující o aktuální pozici uživatele v systému. Zde je uveden názorný příklad drobečkového menu pro první přednášku ze Signálů a systémů v roce 2008:

Domů » 2008 - ISS Signály a systémy » ISS - 1. přednáška, 26. 9. 2008

K přednášce je umožněno zadat klíčová slova, případně krátký textový popis. Pokud některá klíčová slova budou obsahovat i jiné přednášky, může být na ně uživatel upozorněn v seznamu příbuzných přednášek.

4.5 XML soubory

Data pro prohlížeč přednášek jsou z velké části uložena v XML souborech. XML jazyk je podobný HTML, nicméně nemá žádné předdefinované značky a jeho syntaxe je přísnější. Je určen pro výměnu dat mezi aplikacemi a pro publikování dokumentů. Umožňuje dobře popsat vnitřní strukturu dokumentu z hlediska jeho obsahu.

Ve webovém prohlížeči přednášek nachází XML soubory využití pro uchování jazykově závislých textových řetězců (jedná se o anglický a český jazyk) a také pro textová data přednášek.

4.5.1 Jazykové rozhraní prohlížeče

Jazykově závislé řetězce jsou načítány z XML souboru s jednoduchou vnitřní strukturou. V prohlížeči existují dvě jazykové lokalizace – česká a anglická. Každá má svůj vlastní soubor. Povinně je potřeba zadat tzv. informační hlavičku umístěnou v tagu <info>, která definuje základní údaje jazykového souboru. Jedná se o zkratku jazyka (tzv. jazykový kód) <language_id> a jeho název <language_name>.

Význam ostatních tagů je přímočarý. Pokud se jedná o kratší jazykově závislý textový řetězec, název tagu odpovídá přímo tomuto řetězci v anglickém jazyce, kdežto u delších řetězců je význam tagu uveden v angličtině v přecházejícím komentáři zapsaném <!-- #... -->. Příklad části jazykového souboru pro češtinu:

```
<?xml version="1.0" encoding="utf-8" ?>
<strings>
  <!-- language file information -->
  <info>
    <!--
      the language id of the language used in this file;
      it has to be same as the file name (without the extension)
      eg. for the language file "en.xml" -> en
    -->

    <language_id>cz</language_id>

    <!-- the proper name of the language -->
    <language_name>čeština</language_name>
  </info>

  <!-- ***** LANGUAGE DEPENDENT STRINGS ***** -->

  <!-- #Lecture Browser -->
  <title>Prohlížeč přednášek</title>

  <!-- top menu -->
  <contact>Kontakt</contact>
  <faq>FAQ</faq>
  <sign_in>Přihlásit se</sign_in>
  ...
</strings>
```

Není problém tedy prohlížeč jednoduše obohatit o další jazykovou verzi. Pro tento účel by bylo potřeba ještě upravit soubor s nastavením a také rozšířit kód, ze kterého jsou dynamicky sestavovány HTML stránky (např. přidání odkazu na požadovanou lokalizaci).

4.5.2 Uložení dat přednášek

Seznam dostupných přednášek v prohlížeči se nachází ve speciálním souboru (cesta k němu je nastavitelná ve třídě *Settings.php*), který obsahuje dvojice tvořené unikátním číslem přednášky <id> a cestou k jejímu hlavnímu datovému souboru <datafile>.

```

<?xml version="1.0" encoding="utf-8"?>
<data>
  <lecture>
    <id>1</id>
    <datafile>data/2008-winter/iss/01/iss01.xml</datafile>
  </lecture>

  <lecture>
    <id>2</id>
    <datafile>data/2008-winter/iss/02/iss02.xml</datafile>
  </lecture>
  ...
</data>

```

Každá přednáška by měla být pro přehlednost uložena ve vlastním adresáři. Přístupovým bodem je hlavní datový XML soubor přednášky, který obsahuje informace s ní spjaté a případně odkazy na další soubory (slajdy, titulky, transkript aj.). Význam jednotlivých tagů je z následujícího ukázkového souboru snadno pochopitelný.

```

<?xml version="1.0" encoding="utf-8"?>
<data>

  <category>
    <type>lecture</type>
    <year>2008</year>
    <term>winter</term>
    <subject>Signály a systémy</subject>
    <subject_id>ISS</subject_id>
  </category>

  <title>3. přednáška, 8. 10. 2008</title>
  <description>Spojité signály a jejich frekvenční popis: periodické - Fourierova řada (FŘ), koeficienty. Neperiodické - Fourierova transformace (FT), spektrální funkce. Spektra typických signálů. Energie signálu - Parsevalův teorém.</description>
  <keywords>fourierova řada, fourierova transformace, jednotkový impuls, spektrální funkce, zpětná fourierova transformace, parsevalův teorém</keywords>

  <datetime>
    <recorded>8. 10. 2008, 17:00 - 19:59</recorded>
    <added>9. 10. 2008 09:43</added>
  </datetime>

  <author name="Honza Černocký" url="http://www.fit.vutbr.cz/~cernocky/"
email="cernocky@fit.vutbr.cz" />

  <files>
    <video src="iss03.flv" length="02:31:34" resolution="768x576 px" size="396 MB" />
    <transcript src="iss03_transcript.xml" />
    <subtitles src="iss03_subtitles.xml" />
    <slides src="iss03_slides.xml" />

```

```

<thumbnail src="iss03.jpg" />
<timeline src="timeline/timeline.png" thumbnail="timeline/timeline_thumbnail.png" />
</files>

<links>
<a href="http://www.fit.vutbr.cz/study/courses/ISS/public/">Stránky kurzu ISS</a>
<a href="http://en.wikipedia.org/wiki/Fourier_series">Wikipedia - Fourier Series</a>
<a href="http://en.wikipedia.org/wiki/Fourier_transform">Wikipedia - Fourier
Transform</a>
<a href="http://mathworld.wolfram.com/FourierTransform.html">MathWorld - Fourier
Transform</a>
</links>

</data>

```

Syntaxe dalších XML souborů, ze kterých jsou načítána data pro komponenty Transkript, Videozáznam a Slajdy, je vysvětlena přímo v příslušných částech podkapitoly Komponenty (viz 4.6).

4.6 Komponenty prohlížeče

Webový prohlížeč přednášek se skládá z komponent, některé jsou vzájemně synchronizovány podle časové osy přednášky (viz 4.7). Každá komponenta poskytuje uživateli určitou informační hodnotu. Cílem této podkapitoly je seznámit čtenáře s jednotlivými komponentami.

4.6.1 Videozáznam

Tato komponenta slouží pro přehrávání videozáznamů přednášek, které jsou na serveru uloženy ve FLV formátu. Jako přehrávač slouží velice rozšířený JW FLV Media Player (viz 4.10.2.2). Video je doručováno skrze pseudo-streaming umožněný díky spojení zásuvného modulu *mod_flv_streaming* a serveru Lighttpd (viz 4.10.2.1).

K videozáznamu je možnost připojit soubor s titulky, které jsou ve formátu XML založeného na *W3C Timed-Text*¹ specifikaci. Data pro tuto komponentu jsou načítána z hlavního XML souboru přednášky.

```

...
<files>
  <video src="iss03.flv" length="02:31:34" resolution="768x576 px" size="396 MB" />
  <subtitles src="iss03_subtitles.xml" />
  ...
</files>
...

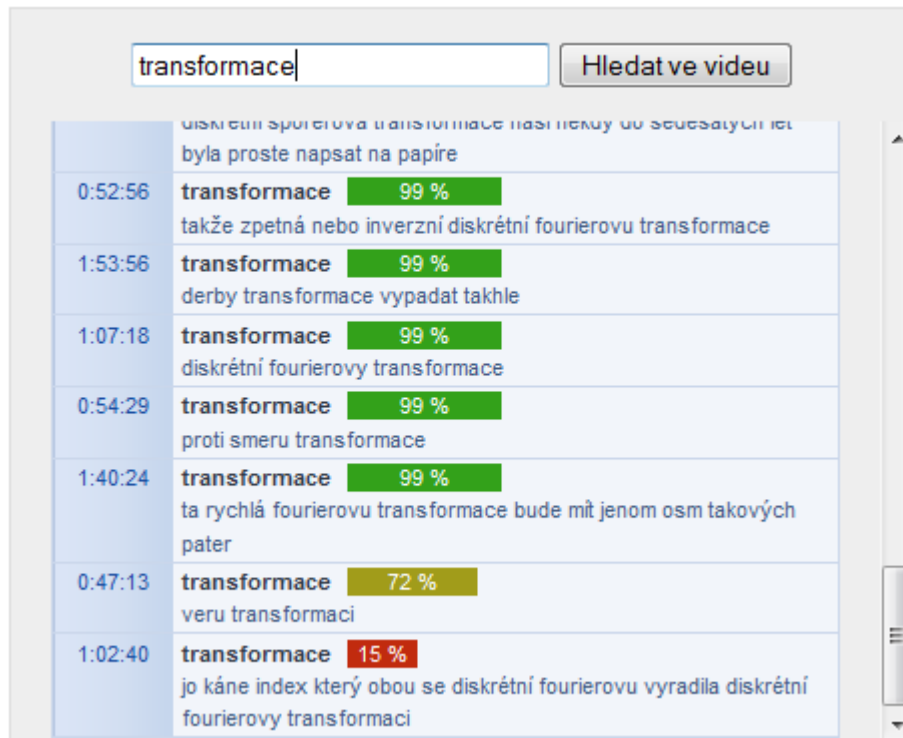
```

Uživatel má možnost přepnout si video do celoobrazovkového režimu, libovolně se ve videu posunovat, měnit jeho hlasitost aj.

¹ <http://www.w3.org/AudioVideo/TT/>

4.6.2 Vyhledávání v audio datech

Skrze tuto komponentu je poskytován uživateli přístup ke speciálnímu serveru, který provádí vyhledávání v audio datech extrahovaných z videozáznamu přednášky. Samotný proces je realizován technologií AJAX, tedy komunikace mezi webovým prohlížečem a serverem probíhá „na pozadí“. To znamená, že uživatel může nadále nedotčeně sledovat přednášku. Pokud hledaný termín existuje, dojde k zobrazení tabulky s výsledky. Ta odkazuje na místa v přednáškách, pro která byl hledaný řetězec rozpoznán. Důležitou informací je tzv. skóre, které udává míru jistoty správnosti rozpoznávače. Doplňkovým údajem je poté textový segment složený ze slov s nejvyšším skóre, ve kterém se hledaný termín může nacházet.



The screenshot shows a search interface with a search bar containing the text 'transformace' and a button labeled 'Hledat ve videu'. Below the search bar is a table of search results. Each row includes a timestamp, the search term 'transformace', a percentage score, and a snippet of text from the video transcript.

Timestamp	Score	Text Snippet
0:52:56	99 %	transformace také zpetná nebo inverzní diskretní fourierovu transformace
1:53:56	99 %	transformace derby transformace vypadat takhle
1:07:18	99 %	transformace diskretní fourierovy transformace
0:54:29	99 %	transformace proti smeru transformace
1:40:24	99 %	transformace ta rychlá fourierovu transformace bude mít jenom osm takových pater
0:47:13	72 %	transformace veru transformaci
1:02:40	15 %	transformace jo káne index který obou se diskretní fourierovu vyradila diskretní fourierovy transformaci

Obrázek 4.1: Vyhledávání v audio datech přednášky

Hledané řetězce spolu s dalšími informacemi je možné nechat automaticky ukládat do souboru, což lze poté využít k analýze funkčnosti této komponenty a získání přehledu, které dotazy na server putují a jak je na ně odpovídáno. Nastavení cesty k tomuto souboru a povolení/zakázání logování probíhá přes soubor *Settings.php*.

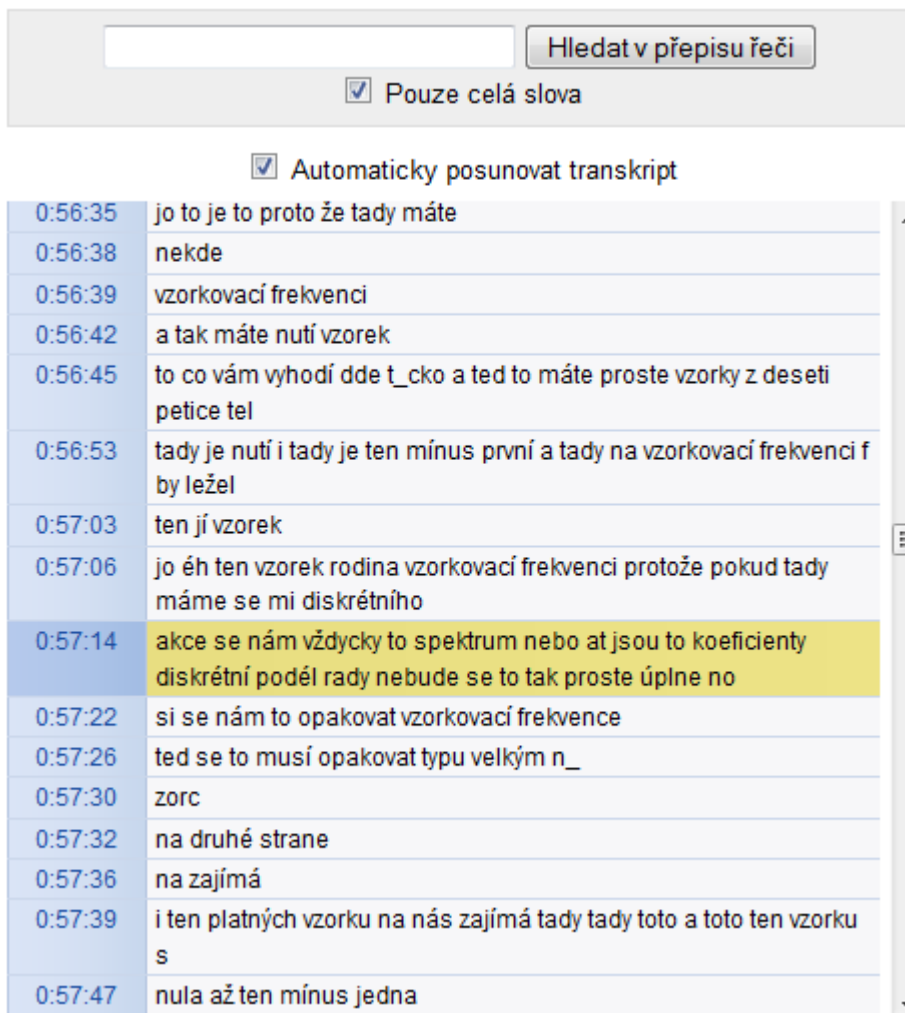
4.6.3 Transkript

Transkriptem se rozumí textový přepis mluveného slova. Tato komponenta zobrazuje v tabulce transkript, který je výhradně generován počítačem, lze tedy předpokládat, že obsahuje chyby. Je složen z nejpravděpodobnějších slov, které jsou získány z výstupu systému pro zpracování řeči. Vytvořit počítačový systém generující naprosto přesný přepis řeči hodinových záznamů přednášek je pro svoji vysokou složitost prakticky nemožné.

Transkript je složen z textových segmentů, kterým přísluší časové body, od nichž byl daný text získán ze zvukového záznamu přednášky systémem pro zpracování řeči. Textovou segmentaci transkriptu lze ovlivnit nastavením příslušného skriptu, který jej sestavuje. Aktuální textový segment je pro snazší orientaci v této komponentě zvýrazněn. Uživatel má možnost se libovolně posunovat v přednášce vybráním určitého segmentu - tím dojde ke vzájemné synchronizaci jednotlivých

komponent (viz 4.7), tedy nastavení odpovídajícího slajdu, pozice videa a označení nového segmentu transkriptu.

Textový přepis přednášky je zpravidla velmi obsáhlý a proto je v této komponentě implementováno vyhledávání, které poskytuje uživateli určitý druh filtrace. Jsou zobrazeny pouze ty segmenty, které obsahují hledaný textový řetězec. Uživatel má možnost omezit vyhledávání na celá slova. Implementace je zajištěna skrze JavaScript, probíhá tedy na straně klienta.



Obrázek 4.2: Komponenta Transkript

Struktura XML souboru, ze kterého jsou načítána data pro komponentu transkript, je triviální. Je složena pouze z nutných dat, jimiž jsou dvojice – časový bod a příslušná část přepisu mluveného slova. Názvy tagů a atributů jsou jednopísmenné z důvodu ušetření místa a také rychlejšího načítání v prohlížeči. Tato úspora se projeví běžně v řádech desítek kB, protože textové přepisy přednášek jsou obvykle složeny ze stovek segmentů.

```
<?xml version="1.0" encoding="utf-8"?>
<data>
<r t="0.15">krásné ráno, pojďme se vrhnout do další přednášky ze signálů a systémů. já se
omlouvám, že na vás budu chrchlat.</r>
<r t="8.35">že bezpečně polomeřeme zeme</r>
...
</data>
```

4.6.4 Slajdy

Tato komponenta umožňuje připojit obrázky slajdů k přednášce a plně je synchronizovat s její časovou osou. Cílem této komponenty je tedy prezentovat slajdy ve vyšší kvalitě, než je k vidění ve videozáznamu. Jelikož je zobrazen seznam všech dostupných slajdů s odpovídajícími časovými body, od kterých jsou slajdy ve videozáznamu promítány, usnadňuje tato komponenta uživateli orientaci v přednášce. Tedy pokud například uživatel potřebuje získat výklad k určitému slajdu, není problém se rychle přenést na patřičné místo ve videozáznamu.

Jak je tomu u harmonického signálu (fázi neuvažujeme): $s(t) = C_1 \cos(\omega_1 t)$? Používáme vzoreček $\cos^2 \alpha = \frac{1 + \cos 2\alpha}{2}$;

$$P_s = \frac{1}{T_1} \int_0^{T_1} [C_1 \cos(\omega_1 t)]^2 dt = \frac{1}{T_1} \int_0^{T_1} C_1^2 \frac{1}{2} (1 + \cos 2\omega_1 t) dt$$

Integrál cosinu je 0 (uvědomte si, že v libovolném množství period funkce cos jsou záporné plochy vyváženy kladnými), takže zůstává:

$$P_s = \frac{1}{T_1} \int_0^{T_1} C_1^2 \frac{1}{2} dt = \frac{1}{T_1} C_1^2 \frac{1}{2} T_1 = \frac{C_1^2}{2}$$

Z toho vyplývá známá poučka o efektivní hodnotě **harmonického** signálu:

$$C_{ef} = \sqrt{P_s} = \frac{C_1}{\sqrt{2}}$$

kterou už jste asi viděli... Pozor! Platí opravdu **pouze** pro harmonický signál!

21

[Zvětšit slajd](#) | Automaticky posunovat tabulku slajdů

1:19:21	33. slajd - Energie a výkon
1:26:50	34. slajd
1:34:26	35. slajd - Neperiodické a periodické signály
1:36:22	36. slajd
1:37:01	37. slajd - Harmonické signály
1:43:42	38. slajd
1:44:35	39. slajd - Výkon periodických signálů
1:48:14	40. slajd
1:54:22	41. slajd - Harmonické signály s diskretním časem
1:59:53	42. slajd
2:02:51	41. slajd - Harmonické signály s diskretním časem
2:04:00	42. slajd
2:05:27	43. slajd
2:07:31	44. slajd - Některé velmi zajímavé a užitečné signály - diskretní
2:11:35	45. slajd
2:12:45	46. slajd - Některé velmi zajímavé a užitečné signály - spojité čas

[Zobrazit všechny slajdy](#)

- [Program a organizace kursu](#) [PDF]
- [Základní pojmy o signálech](#) [PDF]

Obrázek 4.3: Komponenta Slajdy

Název slajdu je složen z unikátního čísla, které je automaticky přiděleno, a volitelně z textového řetězce, jenž blíže specifikuje daný slajd. Pokud je některý slajd zobrazen opětovně, je jeho název odsazen a zobrazen kurzívou. Pokud se jedná o obrázek slajdu vzniklý extrakcí přímo z videozáznamu, je v tomto případě jeho název podbarven modrou barvou. K této komponentě je možno rovněž připojit jakékoliv odkazy, což nachází uplatnění pro umístění přednáškových podkladů.

Pro tuto komponentu jsou data načítána ze speciálního XML souboru, na který je odkazováno z hlavního XML souboru přednášky tagem <slides> (viz 4.5.2). Z hlediska obsahu lze strukturu XML souboru pro slajdy rozdělit do dvou částí. Jedná se o část, která slouží pro připojení odkazů na soubory. Ta je uzavřena v tagu <links> a syntaxe zápisu je v tomto případě stejná jako u komponenty Odkazy (viz 4.6.5). Tou druhou je část pro připojení obrázků jednotlivých slajdů a nachází se v tagu <slides>. Jednotlivé slajdy jsou definovány tagem <slide>. Atribut start udává v sekundách, kdy má dojít ke změně slajdu, src specifikuje cestu k obrázku slajdu, thumbnail k jeho obrázkovému náhledu a extracted signalizuje, že slajd byl automaticky extrahován z videozáznamu. Ukázkový příklad XML souboru pro komponentu slajdy:

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

  <links>
    <a href="slides/documents/01-org_motiv.pdf">Program a organizace kursu</a>
    <a href="slides/documents/02-zakl_sig.pdf">Základní pojmy o signálech</a>
  </links>

  <slides>
    <slide start="5" src="slides/slide01.png" thumbnail="slides/thumbnails/slide01.png"
caption="Signály a systémy (ISS) - program a organizace kursu" />
    <slide start="162" src="slides/slide02.png" thumbnail="slides/thumbnails/slide02.png"
caption="Webové stránky, organizace kursu, kdo vás bude učit?" extracted="extracted" />
    ...

  </slides>

</data>
```

Data pro tuto komponentu mohou být získávána z velké části automaticky díky systému pro automatické zpracování slajdů (viz 2.4.2).

4.6.5 Odkazy

Tato komponenta slouží pro připojení odkazů na zajímavé materiály na internetu. Tedy typickým využitím může být například směrování uživatele na další informace tématicky spojené s přednáškou. Je možné rovněž přiložit odkazy na soubory. Hlavní formáty souborů jsou automaticky rozpoznány a uživatel je o nich informován zkratkou v hranatých závorkách (např. [PDF], či [MP3]).

Odkazy jsou k přednášce přiloženy uvnitř tagu <links>, který se načítá z hlavního souboru přednášky. Jejich počet je neomezen. Syntaxe zápisu těchto odkazů je identická s HTML. Příklad zápisu takovýchto odkazů:

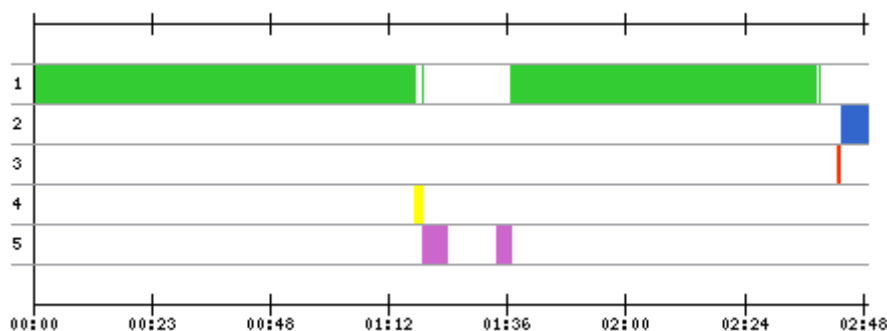
```
<links>
  <a href="http://en.wikipedia.org/wiki/Fourier_transform">Wikipedia: Fourier Transform</a>
  <a href="http://www.falstad.com/fourier/">Fourier Series Applet</a>
</links>
```

4.6.6 Informace o přednášce

Cílem této komponenty je zobrazit uživateli doplňující informace o přednášce. Jedná se o datum a čas pořízení přednášky, rozlišení a velikost videozáznamu. Tyto informace jsou načítány z hlavního XML souboru přednášky.

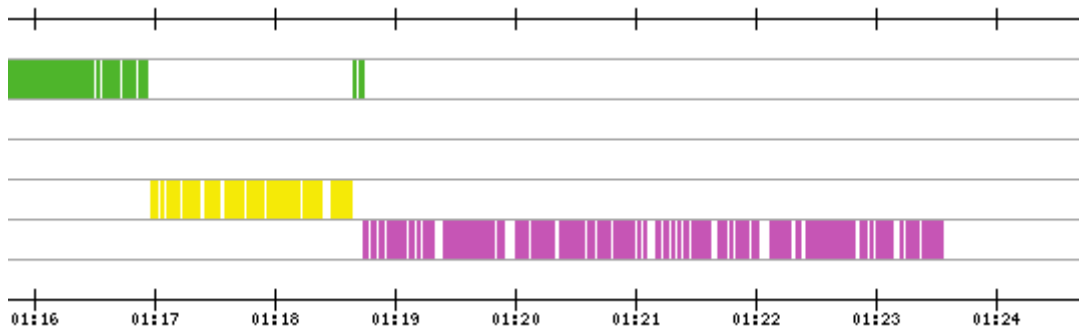
4.6.7 Časová osa přednášky

Časová osa přehledně zobrazuje podíl jednotlivých mluvčích během přednášky. Osa X reprezentuje čas videozáznamu, který je udáván ve formátu [hh:mm]. Osa Y pak náleží jednotlivým mluvčím. Toto znázornění je užitečné zejména pro ty přednášky, na kterých je více mluvčích. Tím lze značně urychlit nalezení úseku pro požadovaného mluvčího.



Obrázek 4.4: Časová osa přednášky – zobrazení pro celou přednášku

Při větším rozlišení lze rovněž dobře určit místa, kdy nikdo nemluvil. Lze se tak například jednoduše přehoupnout přes přestávku v průběhu přednášky. Obrázek s vyšším rozlišením je otevřen ve vyskakovací okně kliknutím na svůj náhled v prohlížeči.



Obrázek 4.5: Časová osa přednášky – zobrazení minutu po minutě

Tato výstupní zobrazení jsou pro prohlížeč vytvořena v průběhu zpracování přednášky, tedy před přidáním dat do prohlížeče. Prohlížeči je dodána dvojice obrázků rozdílného rozlišení. V případě toho většího se v grafu každá sekunda rovná jednomu obrazovému bodu. Zatímco měřítko menšího (tzv. náhledu) je dynamicky upravováno tak, aby výsledná šířka obrázku byla vždy 440 px. Tato velikost je dána návrhem vzhledu stránky. Časová osa se připojuje v hlavním souboru přednášky přes tag <timeline>.

```
<files>
  <timeline src="timeline/timeline.png" thumbnail="timeline/timeline_thumbnail.png" />
</files>
```

4.6.8 Příbuzné přednášky

Tato komponenta slouží k zobrazení náhledů a odkazů na nejbližší podobné přednášky, které jsou spolu tématicky spjaty.

4.7 Synchronizace komponent

Některé komponenty jsou vzájemně synchronizovány podle časové osy přednášky. To znamená, že pokud se změní slajd ve videozáznamu, dojde ke změně slajdu i v odpovídající komponentě prohlížeče. Ta umožňuje zobrazit slajdy ve vyšší kvalitě, zatímco na videu je často problém přečíst jakýkoli text menšího fontu. Rovněž je s časovou osou přednášky synchronizován textový přepis řeči. Tím je myšleno, že je zvýrazněn ten segment transkriptu, který je zrovna aktuální. To samé platí pro titulky, které mohou být volitelně součástí videozáznamu. Čas přednášky lze měnit i vybráním určitého slajdu, tím dojde k přesunu na místo přednášky, na kterém byl daný slajd probírán. To samé platí i pro textový přepis řeči, kdy je možno zvolit textový segment a tím se přenést na místo ve videozáznamu, ze kterého byl zpracován.

4.8 Uživatelské rozhraní

Webové uživatelské rozhraní je oproti rozhraní klasických desktopových aplikací poněkud omezené. Nicméně s využitím různých funkcí jazyků JavaScript, Java, DHTML, Flash lze dosáhnout určitých typicky aplikačních vlastností jako kreslení na obrazovku, přehrávání audio/videoa, přístup k myši, klávesnici, metoda *drag and drop* aj. V poslední době se často začínají používat technologie, které umožňují spolupráci skriptů na klientské straně se serverovou částí aplikace [1]. Zajímavou technikou je Ajax (*Asynchronous JavaScript and XML*), která kombinuje HTML, JavaScript a rozhraní XMLHttpRequest, čímž je dána klientským skriptům možnost načítat informace ze serveru, bez nutnosti obnovovat celou stránku, což vytváří určitý dojem interaktivity.

Stránky webového prohlížeče přednášek jsou sestavovány dynamicky s využitím jazyka PHP. Výsledkem je kód stránek odpovídající standardu XHTML 1.0 Strict. Jejich stylpis je načítán z externích CSS souborů. Grafická úprava stránek je vedena střídavě s převahou odstínů modré a šedé barvy. Rozhraní webového prohlížeče přednášek je stavěné na myšlence reprezentovat jednoduchost a přehlednost s tím, že nový uživatel si na web rychle zvykne a nebude mít problém nalézt požadované informace během krátkého času. Stránky jsou vybudovány pro monitory s rozlišením 1024x768 px a výše.

Skriptovací jazyk JavaScript je použit zejména pro zajištění synchronizace jednotlivých komponent, komunikaci s flash video přehrávačem skrze zasílání tzv. události a pro vyhledávání v rámci transkriptu přednášky. S využitím technologie Ajax je implementováno vyhledávání v audio datech v rámci jedné přednášky. Tedy na server je zaslán požadavek, server odpoví a část webové stránky se překreslí dle získaných informací. Mezitím může uživatel nerušeně pokračovat ve sledování přednášky.



Audio data Hledat



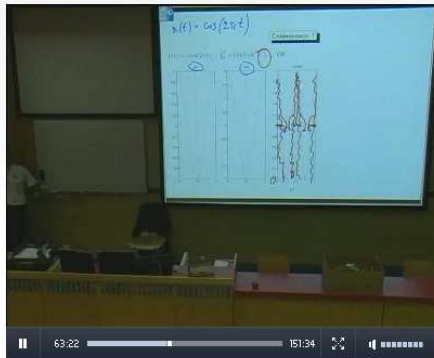
Nacházíte se: Domů » 2008 - ISS Signály a systémy » ISS - 3. přednáška, 8. 10. 2008

Počet přednášek: 7

3. přednáška, 8. 10. 2008

ISS Signály a systémy

Přidáno: 9. 10. 2008 09:43, Autor: Honza Černocký, Délka: 02:31:34



- Slajdy

Cosinusovce ?

$b(t) = 2 \cos(2\pi t)$ $\int_0^1 s(t)b(t)dt = 1$... je podobný

17

- Hledání v audio

vzorek Hledat ve videu

0:58:14	vzorek	99 %
2:10:10	vzorek	99 %
0:41:18	vzorek	99 %
0:57:04	vzorek	99 %
2:16:08	vzorek	99 %
0:57:07	vzorek	99 %
0:54:09	vzorek	96 %
0:18:44	vzorek	88 %
2:21:17	vzorek	74 %
1:59:25	vzorek	7 %
1:14:05	vzorek	2 %
1:57:24	vzorek	1 %
0:27:17	vzorek	0 %
0:55:51	vzorek	0 %
2:13:43	vzorek	0 %
2:03:50	vzorek	0 %

Zvětšit slajd | Automaticky posunovat tabulku slajdů

0:36:21	11. slajd - Otočené souřadnice
0:40:04	12. slajd - Jak to zobrazit pro vícerozměrný prostor?
0:48:44	13. slajd - A obecně v jakoliv rozměrném prostoru...
0:52:02	14. slajd - Jaké chceme báze
0:54:02	12. slajd - Jak to zobrazit pro vícerozměrný prostor?
0:55:54	14. slajd - Jaké chceme báze
0:57:43	15. slajd - Ale signál a báze můžou být klidně funkce!
0:59:26	16. slajd - Stejněměrnému signálu?
1:01:33	17. slajd - Cosinusovce?
1:06:29	18. slajd - 2x rychlejší cosinusovce?
1:07:45	17. slajd - Cosinusovce?
1:08:05	18. slajd - 2x rychlejší cosinusovce?
1:09:57	19. slajd
1:11:15	20. slajd - Tož s obojím...
1:13:31	21. slajd - To už je dobré, ale my máme raději ... Komplexní exponenciály!
1:15:46	22. slajd - Výpočet koeficientů FR

Zobrazit všechny slajdy

Fourierova řada [PDF]

- Přepis řeči

fourier Hledat v přepisu řeči

Pouze celá slova

Hledáno: **fourier**

Počet výskytů/zobrazených segmentů: 73/70

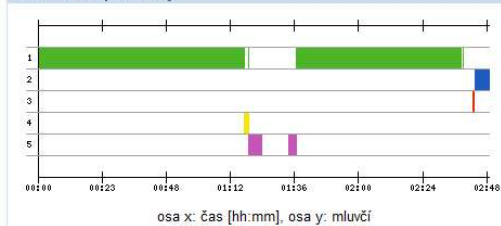
[Zobrazit celý transkript](#)

0:01:02	fourierovu radu fourier ovou transformaci
0:02:52	znamená něco diskretní fourier ovy rade prostě budeme pokračovat dál tom jak se frekvence analyzují
0:03:02	při budete vedet jak se to celé chováme frekvenci a dokončime diskretní fourier ovou radu
0:03:36	a prakticky pomocí se diskretní fourier ovy transformace
0:10:19	o boji což vidíme ted za chvilku treba aukce diskretní fourier ovu tady
0:11:17	fourier ovu transformaci
0:11:20	a tedka se dostáváme k e do diskretní fourier ovy rady
0:15:09	jsi dávno pred mesicem fourier ovu
0:16:37	ted jen tak pekne diskretní fourier ovy rady
0:16:56	a gotické diskretní fourier ovy rady budu taky periodické znamená námi prakticky
0:18:03	jo takže tady tohleto jsou dva vzorecky toto je výpočet koeficientu diskretní fourier ovy rady
0:18:15	na tedka si podle otce diskretní fourier ovy tady neco říct
0:23:23	jako fourier ovu radu nejakých eh posunu v císel
0:23:34	já bych to nehodnotil tady tomuhle nakreslit koeficienty fourier ovu tady
0:23:51	koeficienty fourier ovy rady

+ Odkazy

+ Informace o přednášce

- Časová osa přednášky



- Příbuzné přednášky

2. přednáška, 3. 10. 2008
Signály a systémy
Přidáno: 7. 10. 2008 14:25

4. přednáška, 15. 10. 2008
Signály a systémy
Přidáno: 16. 10. 2008 15:37

Obrázek 4.6: Webový prohlížeč přednášek (stránka pro přehrávání přednášky)

Základním a společným prvkem celého webu prohlížeče přednášek je tzv. hlavička stránek. Uprostřed ní se nachází vyhledávací pole, skrze které je možno provádět vyhledávání v audio datech všech videozáznamů přednášek. Vpravo od něj se nachází ikony vlajek, které slouží pro nastavení jazyka prostředí – angličtina/čeština.

Jednotlivé komponenty jsou umístěny do tzv. rozbalovacích boxů. To znamená, že pokud uživatel nechce mít zobrazenou danou komponentu, může si její obsah jednoduše skrýt kliknutím na hlavičku boxu, která je označena názvem komponenty a symbolem – (schovat) nebo + (zobrazit). Tím dojde ke schování webového obsahu komponenty. Opětovným kliknutím na hlavičku boxu dojde ke zobrazení komponenty. Stav jednotlivých komponent je ukládán do *Cookies*, tedy po znovunačtení stránky je zachováno nastavení viditelnosti jednotlivých komponent. Toto se týká i situace, kdy uživatel přejde na jinou přednášku. Ne všechny komponenty mohou mít pro uživatele požadovanou informační hodnotu a jejich schování ušetří na stránce místo ve vertikálním směru. Pro usnadnění schování/rozbalení všech komponent najednou jsou v horní části stránky tlačítka, která se o požadovanou akci postarají.

Komponenty jsou rozděleny do dvou panelů – levého a pravého. Hlavní komponentou jakékoliv stránky přednášky je videozáznam, pro jehož přehrávání je použit volně dostupný JW FLV Media Player. Tento přehrávač je standardně umístěn v levém panelu. Video může být doplněno o titulky, které jsou získávány z aktuálního segmentu transkriptu. Na stejné horizontální úrovni se vpravo nachází aktuální slajd. Kliknutím na obrázek slajdu dojde k aktivaci tzv. *pop-up window* (vyskakovací okno), ve kterém je zobrazen slajd ve vyšším rozlišení. Vyskakovací okno je použito rovněž pro zobrazení všech slajdů, či detailního zobrazení časové osy přednášky. Pod tímto obrázkem je uveden seznam všech slajdů synchronizovaných s videem a čas, kdy ke změně slajdu dojde. Spodní část tohoto boxu náleží odkazům na soubory se slajdy. Ve většině případů jimi bývají dokumenty ve formátu PDF či PPT. Pod videopřehrávačem je umístěn box, ve kterém lze vyhledávat v audio datech přednášky. Odesláním dotazu dojde k pokusu o spojení se speciálním serverem pro vyhledávání v audio datech. Pokud jsou nalezeny výsledky, je uživateli zobrazena tabulka s odkazy na patřičná místa ve videozáznamu přednášky. Pod tímto boxem se nachází transkript mluveného slova, v němž je možno vyhledávat. Označení aktuálního segmentu transkriptu či slajdu je prováděno barevným zvýrazněním. Ostatní komponenty jako Odkazy, Informace o přednášce, Časová osa přednášky či Příbuzné přednášky jsou k dispozici v rozbalovacích boxech v pravém panelu.

4.9 Ukázková data

Pro otestování webového prohlížeče přednášek byla zvolena skupina 13ti přednášek kurzu Signály a systémy v roce 2008. Jejich videozáznamy spolu se základními informacemi byly staženy z video serveru¹ Fakulty informačních technologií VUT v Brně. Datový tok těchto souborů je oproti výstupu z kamery zmenšen, což se však negativně projevuje na obrazová kvalitě. Častým problémem je poté přečíst obsah slajdů s menšími fonty. Pro použití aplikace provádějící automatické zpracování slajdů z videozáznamu mi byly poskytnuty odpovídající záznamy maximální možné kvality. K přednáškám byly stáhnuty podklady v PDF souborech dostupné na domovské stránce kurzu Signály a systémy.

Pro prvních 9 přednášek byly slajdy zpracovány kompletně ručně. K těmto slajdům byly taky přiřazeny názvy extrahované z jejich nadpisů. S videozáznamem však nebyly synchronizovány podklady použité pro numerická cvičení (nejedná se totiž o klasické slajdy). Nicméně tyto a také ostatní podklady jsou v prohlížeči uživateli dostupné ke stažení jako PDF soubory. Software pro automatickou synchronizaci slajdů s videozáznamem byl nasazena na 10. – 13. přednášku. Jako

¹ <http://video1.fit.vutbr.cz>

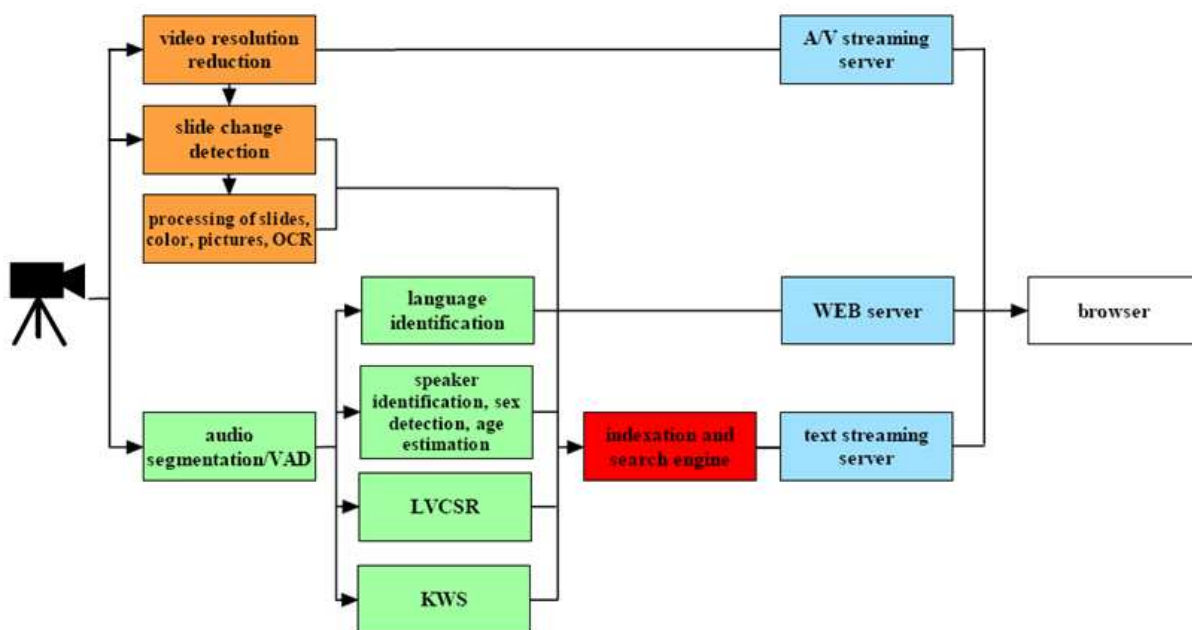
slajdy jsou použity buď obrázky v PNG souborech extrahované z PDF souborů sloužících jako podklady pro přednášky, anebo přímo extrahované slajdy v případě, že algoritmus extrahovaný slajd nepřihřadil žádnému PNG souboru.

4.10 Technické řešení

Webový prohlížeč přednášek potřebuje ke svému běhu Lighttpd server se zásuvným modulem *mod_flv_streaming*. Stránky jsou sestavovány prostřednictvím PHP skriptů a je dodržena specifikace *XHTML 1.0 Strict*¹. Pozice jednotlivých elementů stránky, jejich vzhled, rozměry a další vlastnosti jsou nastavovány skrze CSS soubory. Vyhledávání v aktuálním transkriptu přednášky, synchronizace komponent, změna vlastností elementů aj. je řešena jazykem JavaScript. Textová data jsou uložena v XML souborech. Vyhledávání v audio datech v rámci jedné přednášky je uskutečněno technologií AJAX. Tedy komunikace se serverem probíhá na pozadí a je překreslena pouze část komponenty pro vyhledávání v audio datech, nikoliv celá stránka

4.10.1 Přidávání nových dat do prohlížeče

Z přednášky jsou pořizovány videozáznamy a rovněž jsou k dispozici soubory se slajdy, což představuje hlavní zdroj dat pro prohlížeč. Tato data jsou pak následně zpracována různými rozpoznávači. Výsledek je dodán prohlížeči, který zobrazuje přednášku v multimediální podobě s využitím výstupů rozpoznávačů a jiných programů, které se podílejí na zpracování dat.



Obrázek 4.7: Návrh multimediálního prohlížeče [převzato z 5]

Proces přidávání dat do prohlížeče se sestavuje tedy z několika kroků. Nejprve je potřeba videozáznam přednášky převést do FLV formátu a obohatit jej o tzv. startovní body, které slouží pro zajištění interaktivity při přehrávání Flash videa. Na stopu videozáznamu se aplikují nejrůznější řečové technologie. Výstupem je pak například transkript, který je pro prohlížeč přednášek složen z nejpravděpodobnějších rozpoznávaných slov. Jeho strukturu je však nezbytné upravit skripty tak, aby

¹ <http://www.w3.org/TR/xhtml1/>

vyhovovala požadavkům XML souboru pro komponentu *Transkript*. Ze získaného přepisu řeči je také možno nechat vygenerovat obrázky pro komponentu *Časová osa přednášky*, která zobrazuje podíl jednotlivých mluvčích v průběhu přednášky a místa, ve kterých nikdo nemluvil. Dále je vytvořen XML soubor pro titulky připojitelné ke komponentě *Videozáznam*.

Data pro komponentu *Slajdy* jsou získávána díky systému pro zpracování slajdů z videozáznamu přednášek (viz 2.4.2). Výstupem je XML soubor, který je však potřeba dále upravit skriptem, který z něj vytvoří potřebný XML soubor pro komponentu *Slajdy* a připraví jednotlivé obrázky slajdů a jejich náhledy. Další data přednášky týkající se jejího názvu, zařazení, času a data přidání, vytvoření aj. jsou do systému zadávána ručně.

Jednotlivé soubory a informace pak multimediální prohlížeč prezentuje uživateli v přehledné a ucelené formě.

4.10.2 Použité aplikace třetí strany

Tato podkapitola stručně prezentuje aplikace, které zabezpečují chod webového prohlížeče přednášek, anebo nacházejí využití pro přípravu potřebných dat.

4.10.2.1 Lighttpd

Lighttpd¹ je webový server, jenž klade důraz na bezpečnost, rychlost, flexibilitu a kompatibilitu se standardy. Je navržen a optimalizován pro vysoce zátěžová prostředí. Lighttpd spadá pod BSD licenci. Jeho nízká paměťová náročnost, malé zatížení procesoru a rychlost z něj dělají ideální řešení pro servery, které jsou velmi zatížené. Mezi nejznámější uživatele patří například Youtube, MySpace a Wikipedia.

Pro tento projekt nachází Lighttpd své uplatnění v kombinaci se zásuvným modulem *mod_flv_streaming*, který díky operaci *seek* umožňuje poskytovat Flash video soubory (viz 2.1.1.8) od jakéhokoliv startovního bodu (*cue point*). Jedná se tedy o formu tzv. pseudo-streamingu založeného na progresivním stahování (viz 2.2.6.2). Pro potřeby tohoto projektu je Lighttpd nasazen na operačním systému Linux.

4.10.2.2 JW FLV Media Player

JW FLV Media Player² je nejrozšířenější internetový a velice flexibilní multimediální přehrávač. Zvládne přehrát jakýkoli formát, který je podporován Adobe Flash přehrávačem (FLV, MP4, MP3, AAC, JPG, PNG aj.). Nechybí rovněž podpora RTMP, HTTP, streamingu, titulků, zásuvných modulů, nastavení vzhledu aj. Výhodou je přímá podpora serveru Lighttpd ve spojení se zásuvným modulem *mod_flv_streaming* a podpora obsluhy událostí zasílaných přes JavaScript.

Pro prohlížeč přednášek je JW FLV Media Player použit jako přehrávač videozáznamů. Je upraveno jeho nastavení a modifikován vzhled speciálním souborem. Titulky jsou zobrazovány skrze speciální plugin *Accessibility*. Podpora ovládání přes JavaScript nachází své uplatnění například v situacích, když se uživatel chce přesunout na určité místo ve videozáznamu příslušející danému slajdu či textovému segmentu transkriptu. Kliknutím na patřičný odkaz dojde k zaslání zprávy přehrávači, že má začít přehrávat videozáznam od jistého časového bodu.

¹ <http://www.lighttpd.net>

² <http://www.longtailvideo.com>

4.10.2.3 Mencoder

Mencoder¹ je volně dostupný multiplatformní konvertor video souborů s rozhraním příkazové řádky. Je postaven na stejném kódu jako přehrávač Mplayer. Jeho uplatnění pro prohlížeč přednášek spočívá v převádění MPEG/AVI videozáznamů do formátu FLV. Program Mencoder byl pro sadu ukázkových videozáznamů přednášek volán s těmito parametry:

```
mencoder $input_video_file -ofps 25 -o $output_video_file -of lavf -
oac mp3lame -lameopts abr:br=56 -srate 22050 -ovc lavc -lavcopts
vcodec=flv:keyint=25:vbitrate=$vbitrate:mbd=2:mv0:trell:v4mv:cbp:last
_pred=3
```

4.10.2.4 Yamdi

YAMDI² je zkratkou pro *Yet Another MetaData Injector*. Jedná se o multiplatformní konzolovou aplikaci napsanou v jazyce C, která umožňuje rozšířit FLV soubory o metadata (konkrétně o tzv. *onMetaData event*). Tato utilita je vhodná zejména pro práci s velkými FLV soubory, což je právě případ videozáznamů přednášek. Na rozdíl od jiných podobných řešení video soubor není kompletně načten do paměti. Aplikace je proto méně náročná na paměťové prostředky a je velmi rychlá [26].

Pro prohlížeč přednášek přináší vložení metadat do FLV souboru s sebou možností se ve videozáznamu posunovat díky tzv. startovním bodům (*cue points*). Ve spojení s modulem pro streaming FLV videa je pak možno zahájit přehrávání videozáznamu od jakéhokoliv startovního bodu. Použití pro potřeby prohlížeče přednášek je triviální, je nutné zadat pouze vstupní a výstupní FLV soubor.

```
yamdi -i $input_video_file -o $output_video_file
```

4.10.2.5 Office Convert PDF to Image

Office Convert PDF to Image³ je aplikace, která umožňuje převést jednotlivé stránky PDF dokumentů do obrázků rozličných formátů. Pro prohlížeč přednášek má toto význam při zpracování souborů s podklady přednášek, kdy je potřeba připravit obrázky pro komponentu slajdy. Tyto obrázky jsou poté synchronizovány s videozáznamem přednášky. Aplikací umožňujících převod z PDF do různých grafických formátů je celá řada. Tato ale vyniká svoji jednoduchostí a i základní verze, která je zdarma, dostačuje.

¹ <http://www.mplayerhq.hu>

² <http://yamdi.sourceforge.net>

³ <http://www.officeconvert.com>

5 Hodnocení

Prohlížeč přednášek byl vybudován na množině základních funkcionálních požadavků. Vychází z řešení již několika existujících prohlížečů (viz kapitola 3). Největší porce inspirace byla čerpána z řešení portálu Videlectures.net a prohlížeče MBrowser, který je vyvíjen na FIT VUT v Brně. Videlectures.net představuje zajímavě pojatý webový prohlížeč obsahující stovky různých přednášek pořizovaných na seminářích, konferencích aj. MBrowser je multiplatformní řešení napsané v jazyce C++ sloužící pro lokální přehrávání přednášek, zatímco zde navržený prohlížeč je postaven na webových stránkách, čímž je dostupný přímo přes jakýkoliv internetový prohlížeč. Inspirace pro grafické uživatelské rozhraní a rozmístění komponent pak vychází z internetových video portálů jako Videlectures.net a jiných.

Při vývoji bylo nutno překonat rozdíly v implementacích jednotlivých internetových prohlížečů, což značně zpomalovalo vývoj. Prototyp byl testován v nejrozšířenějších prohlížečích a měl by podobně fungovat alespoň v Internet Explorer verze 6 a vyšší, Firefox, Opera a Google Chrome.

Prohlížeč byl naplněn ukázkovými daty z kurzu Signály a systémy 2008 a je tak možné tuto verzi použít jako podpůrný materiál pro pochopení probírané látky v rámci budoucích ročníků tohoto kurzu. Vytvoření množiny ukázkových dat trvalo značnou dobu, protože bylo potřeba nalézt vhodné postupy. Rovněž nebyl zpočátku k dispozici software pro automatické zpracování slajdů z videozáznamu, proto větší část ukázkových přednášek byla zpracována ručně.

Výsledný produkt prezentuje multimediální možnosti webového prohlížeče přednášek podpořené technologiemi pro zpracování řeči a obrazu. Hodnocení kvality implementovaného prohlížeče bude zejména pak postaveno na reakcích jeho uživatelů. Se zveřejněním verze, která je součástí této práce, se počítá. Webový prohlížeč přednášek rozhodně poskytuje přínos pro studium problematiky probírané v rámci přednášek a usnadňuje cestu potřebnou k nalezení požadované informace. Integrace různorodých materiálů souvisejících s přednáškou do jednoho místa, které je zároveň dostupné přes webové stránky snad z jakéhokoliv počítače připojeného k Internetu, určitě urychlí a ulehčí získávání vědomostí. Vytyčený počáteční cíl, kdy prohlížeč by měl pomáhat uživatelům vsřebávat informace probírané na přednáškách, byl splněn. Pro reálné nasazení webového prohlížeče přednášek v rámci více kurzů je však vhodné současný systém dále rozšířit. Touto tematikou se zabývá následující kapitola 6.

5.1 Možnosti širšího uplatnění prohlížeče

Prohlížeč je navržen s cílem použití pro přednášky. Nicméně jeho využití může být nepochybně mnohem širší. Rozhodně svůj význam může najít při nasazení pro různé konference, meetingy, ze kterých bývá pořizován videozáznam. Aplikace technologie pro vyhledávání v řečových datech na jakýkoliv audio/video materiál má rozhodně velký potenciál. Například gigant Google nasadil v roce 2008 systém pro vyhledávání v řeči na některá videa ze serveru Youtube týkající se předvolební kampaně boje o post amerického prezidenta [25]. Lze tak rychle vyhledat například různá zajímavá témata, o kterých kandidáti mezi sebou diskutovali a tím se dozvědět, jaký má kdo názor na danou problematiku a rozhodnout se, komu dát ve volbách svůj hlas.

6 Rozšíření prohlížeče přednášek

Projekt webového prohlížeče přednášek je poměrně rozsáhlý. Ke svému běhu využívá další aplikace a technologie týkající se zpracování řeči a obrazu. V této diplomové práci byla vytvořena verze prohlížeče, která demonstruje základní požadovanou funkcionalitu a naznačuje cestu, kterou by se následující vývoj mohl ubírat. Uvažuje se o reálném nasazení prohlížeče pro přednášky pořizované na FIT VUT v Brně. Pro tento účel bude rovněž potřeba současné řešení upravit a rozšířit.

Cílem této kapitoly je nasměrovat následující vývoj prohlížeče přednášek a uvést zajímavé vlastnosti či rozšíření, kterými by další verze mohly disponovat.

6.1 Technické řešení prohlížeče

Současný prohlížeč přednášek je postaven na datech načítaných z XML souborů. Bylo by vhodné data z hlavního XML souboru přednášky pro jednodušší manipulaci a ošetření ukládat do databáze. Množina těchto dat by se měla dále rozšířit o různé informace související s přednáškou. Je dále potřeba vyřešit kategorizaci systému, obsah hlavní stránky a poskytnout administrační rozhraní pro přidání/úpravu/smazání dané přednášky. Jelikož ne všechny přednášky by měly být dostupné kterémukoliv uživateli, je potřeba vyřešit otázku uživatelských rolí a příslušných práv. Dále je potřeba zjednodušit proces přidávání nových dat přednášky. Bylo by vhodné také jednotlivé kroky urychlit (například konverze videa, automatická synchronizace slajdů aj.).

Dalším problémem může být hardware současného serveru. Pro zvládnutí velké zátěže bude patrně potřeba jeho konfiguraci vylepšit.

6.2 Vlastnosti prohlížeče

Následuje seznam návrhů několika rozšíření, které by obohatily a zvýšily funkcionalitu webového prohlížeče přednášek. Významným zdrojem pro další rozvoj prohlížeče bude pak odezva jeho samotných uživatelů.

6.2.1 Inteligentní linkování obsahu

Jelikož je automaticky vygenerovaný transkript pouze čistý neformátovaný text, zajímavým rozšířením prohlížeče přednášek by mohl být modul, který by automaticky obohatil holý textový přepis o hyperlinkové odkazy směřující na užitečné stránky. Příkladem by mohly být články encyklopedie Wikipedia¹, jež by poskytovaly odpovídající informace k danému slovu/frázi. Odkazy by však měly být asociovány pouze s vhodnými encyklopedickými termíny jako například „*Fourierova transformace*“ nebo „*Parsevalův teorém*“. Tento systém by samozřejmě měl brát v úvahu skloňování, diakritiku aj. Rovněž by měla být možnost zobrazit samostatně seznam takovýchto odkazů.

6.2.2 Uživatelsky definovaný prohlížeč

Uživatel by mohl mít možnost si přizpůsobit vzhled, rozmístění jednotlivých komponent a další parametry prohlížeče. Uživatelsky definovaný prohlížeč by byl k dispozici jen registrovaným uživatelům. Po jejich přihlášení by byla automaticky načtena uživatelská konfigurace prohlížeče.

¹ <http://www.wikipedia.org>

Na výběr by uživatel měl mít možnost si zvolit styl vzhledu prohlížeče, případně napsat vlastní CSS soubor. Prohlížeč bude postaven na šablonovacím systému, který umožní v návaznosti na CSS soubor snadnou změnu vlastností jednotlivých prvků stránky. Komponenty prohlížeče by měly být vzájemně přemístitelné. Rovněž by měla být možnost povolit/zakázat danou komponentu. Komponenty jsou sice vzájemně provázané, ale mohou fungovat i samostatně.

Pro přihlášeného uživatele by měl prohlížeč zaznamenávat a na vyžádání poté přehledně zobrazit určité uživatelské aktivity (např. zhlédnutá videa, seznam vlastních komentářů atd.). Uživatel by měl možnost si vybrat určitou skupinu kategorií přednášek (oblíbené), a na jakékoliv změny (např. přidání nové přednášky) by mohl být informován například upozorněním na hlavní stránce. Vyhledávání v přednáškách by mělo mít možnost být omezeno pouze na přednášky v oblíbených kategoriích/předmětech.

6.2.3 Oznámení o aktualizaci

Při aktualizaci stránek by měla být uživateli nabídnuta volba „*být informován*“ a to například zasláním emailu, jímž by mohl být například doručován seznam nově přidaných přednášek. Rovněž by se tato služba dala využít pro zasílání oznámení ohledně nového komentáře/aktualizace/rozšíření určité přednášky. Zasílání by mělo být filtrováno s možností vlastního uživatelského nastavení. Upozornění by bylo zasíláno jen pro ty přednášky, které spadají do oblasti uživatelského zájmu.

Nově přidané přednášky, které jsou veřejně přístupné, by měly být zahrnuty do speciálního RSS kanálu přednášek. Další RSS kanál by mohl sloužit pro zprávy týkající se webového prohlížeče přednášek.

6.2.4 Rozšíření typů videozáznamů přednášek

Pro přehrávání videa využívá webový prohlížeč přednášek výhradně technologie Flash. Nicméně uživatel by mohl mít k dispozici ještě volbu použití jiného streaming řešení. Prohlížeč by mohl mít například možnost využívat VLC plugin pro prohlížeče, kterým lze zachytávat videozáznamy přednášek, které jsou vysílány přes RTSP protokol. Na současných video serverech FITu je tento protokol nainstalován a je tak možno shlédnout záznamy příslušnými programy bez nutnosti je stahovat. Nicméně současný VLC Mozilla plugin, či VLC ActiveX prvek neposkytují úplně stabilní řešení.

Rozhodně by také neměla chybět možnost stáhnutí videozáznamu přednášky na počítač uživatele, ideálně by měl být nabídnut videozáznam v různých verzích lišících se velikostí a kvalitou videozáznamu.

6.2.5 Online streaming

Bohužel webový server Lighttpd ve spojení s modulem pro streaming Flash videa nepodporuje živé vysílání. Aby toto bylo možné, bylo by potřeba nasadit opravdový streaming server. Řešení od Adobe je poněkud finančně nákladné, nicméně existuje řešení v podobě RED5 open source Flash serveru, které je zadarmo.

6.3 Návrh dalších komponent

Implementovaný prohlížeč obsahuje řadu komponent, nicméně pro další vývoj by bylo vhodné jejich počet dále rozšířit. Dnes velmi rozšířenou součástí různých portálů je možnost vyjádření názoru uživateli. Tato vlastnost však může být také snadno zneužitelná, proto je nutno k tomuto tématu přistupovat obezřetně.

6.3.1 Komentáře

K přednášce bude možnost zapnout přidávání komentářů. Toto umožní uživatelům pokládat své dotazy, připomínky a vzájemně je sdílet. Komentování přednášky bude omezeno na autorizované uživatele. Nevhodné komentáře bude možno smazat a uživatelé by měli mít možnost na ně jednoduše upozornit administrátora systému. Dále by neměla chybět možnost jednotlivé komentáře hodnotit. Ty, které by získaly velmi negativní hodnocení, by neměly být vůbec zobrazeny, zatímco komentáře s pozitivním hodnocením by byly upřednostňovány před ostatními.

6.3.2 Vylepšený transkript

Opravu nepřesností způsobenou strojovým rozpoznáváním by měl mít možnost provádět autorizovaný uživatel. Vytvořený systém by například mohl být založen na podobném principu jako Wiki – jakékoliv provedené změny by byly zaznamenávány. Pokud by došlo k neadekvátnímu zásahu, dal by se jednoduše vrátit jakýkoliv předchozí stav. Míra kvality textového přepisu řeči by se rovněž dala vylepšit určitými technologiemi pro zpracování přirozeného jazyka.

6.3.3 Slajdy

Jelikož budou slajdy uloženy jako obrázky ve formátu PNG, lze na ně aplikovat technologii OCR (*Optical Character Recognition*) a získaný textový přepis dále zpracovat. Pro určité slajdy by tak bylo možno například automaticky generovat jejich názvy z nadpisů, nebo obecně by se v nich dalo vyhledávat jako v jakémkoliv textu apod.

6.3.4 Ostatní

Mezi další části systému bude patřit možnost vložení souborů, či odkazů v rámci Internetu. K dané přednášce bude možno zobrazit příbuzné přednášky, které budou vybírány z množiny podobných vlastností. Rovněž bude možno přednášku ohodnotit či doporučit. Mohla by být implementována komponenta pro uživatelské poznámky, které by mohly být buď soukromé nebo sdílené v rámci skupiny uživatelů.

7 Závěr

Seznámil jsem se s multimedialním prohlížečem MBrowser, který je vyvíjen na Fakultě informačních technologií VUT v Brně. Porovnal jsem ho s jinými podobně zaměřenými prohlížeči a na základě jejich dovedností jsem vytvořil seznam funkčních požadavků pro webový prohlížeč přednášek, který je předmětem mé diplomové práce.

Důležitým prvkem prohlížeče přednášek je schopnost pracovat s videozáznamy. Z tohoto důvodu jsem věnoval podstatnou část svého času technologiím umožňujícím streaming videa a testoval jejich nejčastější řešení. Seznámil jsem se s řečovými technologiemi, které jsou vyvíjené skupinou Speech@FIT. Informoval jsem se na dostupnost technologií sloužících k automatické extrakci slajdů z videozáznamů přednášek. Dle požadavků byl pro mé potřeby vytvořen speciální software, jenž zvládá extrahovat slajdy z videozáznamů pořízených statickou kamerou, případně umí k těmto extrahovaným slajdům přiřadit odpovídající obrázky získané z podkladů přednášek. Získal jsem videozáznamy v maximální kvalitě a v nezpracované formě, které jsou pořizovány z přednášek. Vyzkoušel jsem různá nastavení pro převod záznamů do Flash videa.

Na základě funkčních požadavků a dostupných technologií jsem navrhnul prohlížeč přednášek. Implementoval jsem požadovanou množinu vlastností prohlížeče. Bral jsem ohled na validitu generovaných stránek a XML souborů. Usiloval jsem o vytvoření jednoduchého a přehledného uživatelského rozhraní. Z tohoto důvodu jsem jednotlivé typy dat rozdělil do komponent, které si uživatel může dle své libosti zobrazit/schovat. Propojil jsem prohlížeč přednášek se serverem pro vyhledávání v audio datech. Otestoval jsem vytvořený systém v nejrozšířenějších webových prohlížečích na operačních systémech Windows a Linux.

Jelikož tento produkt má potenciál reálného nasazení, navrhnul jsem některé vlastnosti a vylepšení, kterými by mohla disponovat příští verze prohlížeče přednášek.

Webový prohlížeč přednášek s ukázkovými daty týkající se kurzu Signály a systémy 2008 je nasazen na serveru a je tak dostupný k vyzkoušení přes jakýkoliv internetový prohlížeč. Zdrojové soubory jsou rovněž k dispozici na přiloženém DVD nosiči. Jeho součástí jsou také aplikace, které jsou použity pro chod webového prohlížeče nebo slouží k přípravě potřebných dat. Elektronická verze diplomové práce je přiložena.

Literatura

- [1] Wikipedia, The Free Encyclopedia.
URL <<http://www.wikipedia.org>>
- [2] Streaming Video using Flash. [cit. 2008-12-20].
URL <<http://www.mediacollege.com/video/streaming/formats/flash.html>>
- [3] Flash video learning guide. [cit. 2008-12-20].
URL <http://www.adobe.com/devnet/flash/articles/video_guide.html>
- [4] Streaming Service for Flash. [cit. 2008-12-20].
URL <<http://www.internapcdn.com/flash/index.html>>
- [5] Černocký, H.: Technologie zpracování řeči pro indexování záznamů přednášek. [online]. 14.11.2006. [cit. 2008-12-26].
URL <http://elearning.cesnet.cz/e107_files/downloads/2_cernocky.pdf>
- [6] Fapšo, M.: Search in speech data, master's thesis, Brno, FIT BUT in Brno, 2007.
- [7] VideoLAN Streaming HowTo. [cit. 2008-12-27].
URL <<http://www.videolan.org/doc/streaming-howto/en/streaming-howto-en.html>>
- [8] VideoLAN Wiki - ActiveX. [cit. 2008-12-27].
URL <<http://wiki.videolan.org/ActiveX>>
- [9] Co je a není kodek. [cit. 2008-12-27].
URL <<http://jech.ic.cz/kodek.php>>
- [10] Kodeky tajemství zbavené. [cit. 2008-12-27].
URL <http://www.tvfreak.cz/art_doc-373A9DA2913B7BD3C125727C00592A37.html>
- [11] Kontejner není kontejner. [cit. 2008-12-29].
URL <http://www.tvfreak.cz/art_doc-7336C842E0DDDE25C125727C0059416E.html>
- [12] Orság, F.: Rozpoznávání hlasu, Brno, UIVT FEI VUT Brno, 2000.
- [13] Digitální kompresní formáty. [cit. 2008-12-29].
URL <<http://jech.webz.cz/formaty.php>>
- [14] Puncman, P.: Použití triangulací pro reprezentaci videa, diplomová práce, Plzeň, FAV ZČU v Plzni, 2008.
- [15] QuickTime Streaming Server. [cit. 2008-12-30].
URL <<http://www.apple.com/quicktime/streamingserver/>>
- [16] The Helix DNA Server. [cit. 2008-12-30].
URL <<https://helix-server.helixcommunity.org/>>
- [17] Windows Media Services. [cit. 2008-12-31].
URL <<http://www.microsoft.com/windows/windowsmedia/forpros/server/server.aspx>>
- [18] Microsoft Silverlight. [cit. 2008-12-31].
URL <<http://www.microsoft.com/SILVERLIGHT/>>
- [19] Kubalík, J.: Multimediální prohlížeč pro přednášky, bakalářská práce, Brno, FIT VUT v Brně, 2007.
- [20] Glass, J. *et al.*: Recent Progress in the MIT Spoken Lecture Processing Project. In *Proceedings of Interspeech'2007*, Antwerp, Belgium, 2007.
- [21] MIT Lecture Browser. [cit. 2009-01-04].
URL <<http://web.sls.csail.mit.edu/lectures/>>
- [22] JFerret - Home. [cit. 2009-01-04].
URL <<http://www.idiap.ch/mmm/tools/jferret>>
- [23] About VideoLectures.net. [cit. 2009-01-05].
URL <<http://videlectures.net/site/about/>>

- [24] Kubalík, J.: Multimedia Browser P.R.A.S.E.. *Ve sborníku EEICT 2008*, Brno, 2008.
- [25] Search what the politicians are saying. [cit. 2009-01-05].
URL < <http://labs.google.com/gaudi>>
- [26] Yamdi - Yet Another MetaData Injector. [cit. 2009-03-17].
URL < <http://yamdi.sourceforge.net>>
- [27] Sumec, S.: Extrakce doplňkových materiálů ze záznamů přednášek. [online]. 14.11.2006.
[cit. 2009-03-18].
URL <http://elearning.cesnet.cz/e107_files/downloads/2_sumec.pdf>
- [28] Sukthankar, R., Stockton, R. G., Mulin, M. D.: *Smarter Presentations: Exploiting Homography in Camera-Projector Systems*, Proceedings of International Conference on Vision, 2001.

Seznam příloh

Příloha 1.

DVD obsahující zdrojové kódy webového prohlížeče přednášek a elektronickou verzi diplomové práce.

Příloha 2.

Online verze prohlížeče přednášek nasazená na serveru a dostupná na této URL:

<<http://pcspeech7.fit.vutbr.cz/mbrowser>>