MASARYKOVA UNIVERZITA
FAKULTA INFORMATIKY

DIPLOMOVÁ PRÁCE

SYSTÉMY NA PODPORU DIGITÁLNÍCH KNIHOVEN (GREENSTONE)

JAKUB ŘEHAN

BRNO 2004

Prohlášení

Prohlašuji, že tato práce je mým původním autorským dílem, které jsem vypracoval samostatně. Všechny zdroje, prameny a literaturu, které jsem při vypracování používal nebo z nich čerpal, v práci řádně cituji s uvedením úplného odkazu na příslušný zdroj.

Poděkování

Rád bych poděkoval RNDr. Miroslavu Bartoškovi, Csc. za ochotu a odbornou pomoc při vedení této diplomové práce. Dále chci poděkovat panu Romanovi Chýlovi za cenné rady a pomoc při lokalizaci rozhraní. Poděkování patří i spoluautorům systému Greenstone, zvláště paní Katherine Don a panu Michaelu Dewsnipovi z univerzity Waikato.

Shrnutí

Tato práce se zabývá volně dostupnými systémy na tvorbu a správu digitálních knihoven se zvláštním zřetelem k systému Greenstone. V teoretické části práce jsou nastíněny některé základní problémy oblasti zpracovávání a uchovávání informací v digitální podobě spolu s popisem volně dostupných nástrojů pro vytváření a zpřístupňování organizovaných sbírek digitálních materiálů. Zvláštní důraz je kladen na popis systému Greenstone.

Praktická část shrnuje výsledky dosažené při řešení zadání diplomové práce spolu s poznatky získanými při používání systému Greenstone. Je představena aplikace Simple Collection Manager, která byla vytvořena jako grafické uživatelské rozhraní pro základní správu sbírek a manipulaci s metadaty. V dalších kapitolách je podrobně rozebrán postup tvorby dvou vybraných ukázkových sbírek, které zároveň slouží pro další vysvětlení principů fungování digitálních knihoven Greenstone. Praktická část také obsahuje shrnutí přínosu práce při zpřístupnění digitální knihovny českým uživatelům a popis dalších možností pro přizpůsobení českému prostředí. Na závěr jsou uvedeny kapitoly zabývající se popisem a vysvětlením dvou klíčových mechanismů, pluginů a maker, používaných systémem Greenstone.

Klíčová slova: Greenstone, digitální knihovna, metadata, dokument, sbírka, archivace, zpřístupňování, interoperabilita

Obsah

1 Úvod

Cílem této práce bylo prozkoumat volně dostupné systémy na podporu digitálních knihoven se zvláštním důrazem na systém Greenstone ([20]). Kromě analýzy a systémového popisu tohoto softwarového produktu bylo úkolem přizpůsobit jej českému uživateli pomocí lokalizace rozhraní a vytvoření některých komponent tak, aby byla digitální knihovna Greenstone snáze použitelná v českém prostředí. Součástí řešení je realizace několika ukázkových sbírek demonstrujících nabízené možnosti řízené správy dokumentů s využitím tohoto nástroje a aplikace pro jednoduchou tvorbu sbírek digitálních materiálů.

V úvodních kapitolách této práce jsou digitální knihovny popsány v kontextu uchovávání informací v historii a současnosti. Kromě definice pojmu digitální knihovny je také poskytnuto srovnání s knihovnami klasickými a vymezení jejich vzájemného vztahu.

Teoretická část práce v úvodu shrnuje některé nejdůležitější pojmy z oblasti digitálních knihoven. Další kapitoly jsou věnovány popisu struktury systému Greenstone a klíčových pojmů a mechanismů používaných pro vytváření a správu sbírek v digitálních knihovnách systému Greenstone. Na závěr teoretické části je uveden přehled dalších volně dostupných systémů, umožňujících vytváření a zpřístupňování rozsáhlých sbírek digitálních objektů.

Praktická část shrnuje výsledky práce a poznatky získané při používání systému Greenstone. V úvodu je popsán nástroj Simple Collection Manager, který byl vytvořen jako grafické uživatelské rozhraní pro základní správu zdrojových materiálů sbírek knihovny. Další kapitoly se zabývají popisem postupu vytváření vybraných ukázkových sbírek a zároveň slouží jako detailnější popis některých důležitých mechanismů fungování knihovny samotné. Praktická část dále shrnuje přínos této diplomové práce při přizpůsobování systému Greenstone českému prostředí. Poslední kapitoly nabízejí nejdůležitější poznatky získané při vytváření a používání dvou klíčových struktur digitální knihovny Greenstone – pluginů a maker. Kromě diplomové práce samotné byl vytvořen také ucelený popis systému Greenstone a jejích důležitých komponent pro české uživatele. Dokument je umístěn na CD přiloženém k této práci.

1.1 Klasické knihovny v digitálním období

V průběhu celé lidské historie patří informace k tomu nejcennějšímu. Jejich uchovávání a předávání je jedním z hlavních úkolů každé generace, protože předpokladem dalšího rozvoje společnosti je možnost stavět na získaných zkušenostech. V průběhu historie se vyvinulo množství způsobů zachování kulturního a intelektuálního dědictví – od nástěnných maleb v jeskyních přes nápisy tesané do kamene až po knihovny uchovávající informace v tištěné podobě. Ačkoliv přístup klasických „kamenných“ knihoven se poslední dobou přizpůsobuje tlaku doby, nemůže obsáhnout informační explozi elektronických zdrojů. Prudký rozvoj informačních technologií, zvláště osobních počítačů a celosvětové sítě internetu, umožňuje stále většímu počtu lidí vystupovat nejen v roli konzumentů informací (jako je tomu u klasických knihoven), ale také v roli jejích tvůrců. Náročný a relativně nákladný postup tvorby a rozšiřování informací v tištěné podobě (autor – recenzent – editor – sazeč – knihkupec – čtenář) zároveň sloužil a slouží jako prvotní filtr třídící kvalitní informace a zabraňující záplavě bezcenných informačních zdrojů. Ve virtuálním světě však žádná taková omezení neexistují – vytvoření dokumentu a jeho zpřístupnění prostřednictvím sítě internet vyžaduje pouze čas. Také motivační faktory autorů se v obou oblastech liší. Zatímco u tištěných materiálů (snad s výjimkou vědeckých publikací) je důležitou motivací zisk, u elektronických dokumentů to může být cokoliv od potřeby se zviditelnit až po prezentaci seriózních informací. Uvedené

odlišnosti jsou jen malou ukázkou rozdílů mezi oběma světy, stačí ale k tomu, aby ukázaly jak diametrálně se musí lišit přístup v obou oblastech.

Nám dobře známé knihovny mají za sebou více než 25 století dlouhou historii, ve které se specializovaly na uchovávání informaci ve fyzické, většinou psané či tištěné, podobě. Během tohoto z hlediska naší civilizace dlouhého období se vyvinuly velice efektivní způsoby ke zpracovávání, pořádání a zpřístupňování dokumentů. Příkladem mohou být rejstříky, fyzická organizace usnadňující hledání podobných položek, v poslední době pak standardy umožňující sdílení knihovnických záznamů a podobně. Současné knihovny uchovávají obrovská množství výtisků nejrůznějších titulů a jsou schopny je zároveň poskytovat veřejnosti. I přesto jsou ale postupem času přehodnocovány některé i poměrně důležité principy stojící v základech knihoven. Dlouhou dobu prosazovaný *Alexandrijský princip* (podle známé knihovny v Alexandrii), prosazující shromažďování všech dostupných materiálů, ustupuje pod tlakem neustálého nárůstu počtu tištěných informačních zdrojů přístupu tématicky úžeji zaměřených sbírek, shromažďujících pouze kvalitní informace. Příkladem aplikování toho principu může být likvidace 200 000 knih ze sbírek veřejné knihovny v San Franciscu v roce 1996 (viz [44], str. 14). Jedním z faktorů nutících knihovny ke změně přístupu a provádění podobných radikálních kroků je finanční hledisko. V rozpočtech knihoven už často částky určené pro udržování budov a skladování existujících výtisků přesahují částky určené pro nákup nových materiálů. Při nikdy nekončícím rozšiřování sbírek pak logicky musí dojít k okamžiku, kdy knihovna začne stagnovat, protože veškeré prostředky půjdou do udržování stávajících materiálů.

Klasické knihovny představují velmi propracovaný systém pro uchovávání informací v tištěné podobě. S nástupem digitálního období se však potýkají se složitými problémy souvisejícími se základními principy jejich fungování. Opuštění Alexandrijského principu se nedotýká jen uvedené změny postoje ke způsobu tvorby sbírek, ale také změny v pojetí archivace informací jako celku. Vynález knihtisku vedl k růstu vlivu knihoven jakožto univerzálních informačních zdrojů. Dlouhou dobu toto pojetí stačilo, ale zvláště s rozvojem nových technologií začalo být náročnější této roli dostát. Knihovny se snažily přizpůsobit rozšířením nabídky služeb – začaly půjčovat zvukové nosiče, poskytovat přístup k internetu a podobně. S masivním nástupem elektronických médií a narůstající záplavou elektronických dokumentů vyvstala potřeba nových způsobů jejich organizace, zpřístupňování a zpracování. Ačkoliv řadu důležitých poznatků a přístupů lze z knihoven převzít, nová oblast obsahuje mnoho unikátních problémů, které je potřeba vyřešit.

1.2 Nástup digitálních knihoven

Stejně jako nástup knihtisku umožnil rozvoj knihoven klasických, nástup informačních technologií a záplava dat v digitální podobě umožňuje vznik a rozvoj knihoven digitálních. Dříve než bude uvedena jedna z mnoha definic digitálních knihoven je třeba zdůraznit, že účelem digitálních knihoven není nahradit ty stávající, „kamenné“. Nejde tedy o to *„spálit doma knihy a sedět při zimních večerech u krbu zabraní do obrazovky monitoru“* ([44], str. 6), ale reagovat na nové potřeby uchovávání informací. Knihovny zůstanou důležitým zprostředkovatelem informací v převážně tištěné podobě, digitální knihovny se snaží stát se podobnými prostředníky v oblasti elektronických dokumentů. Oba světy nejsou odděleny, právě naopak spolu úzce spolupracují. Digitální knihovny využívají mnoho poznatků z oblasti knihovnictví, a kamenné knihovny mohou těžit ze služeb nabízených novými technologiemi – prezentace širšímu okruhu veřejnosti, komfortnější vyhledávání nebo například rychlejší spolupráce s ostatními knihovnami.

Samotný termín digitální knihovna není přesně vymezen. Pro někoho to může být „digitalizovaná knihovna“, tedy elektronický zdroj obsahující digitalizovaný obsah existujících fyzických dokumentů. Ačkoliv uchováváním digitalizovaných materiálů, získaných převodem existujících artefaktů do elektronické podoby (na rozdíl od těch digitálně vytvořených – *born digital* – které už jako digitální vznikají), se digitální knihovny zabývají také, tato interpretace pojmu svádí k představě pouhého archivu. Zásadním rozdílem, který odlišuje i klasické knihovny od pouhých skladů knih, jsou poskytované služby. S narůstajícím množstvím dokumentů narůstá také význam služeb, umožňujících efektivně vyhledávat požadované informace. Bez této možnosti je sebevětší sbírka téměř nepoužitelná, neboť úsilí vynaložené na hledání informace převáží význam informace samotné. Pokud nehledáme nic konkrétního, ale spíše se zajímáme o určitou oblast, oceníme také možnost procházení různých materiálů podle jejich vzájemné souvislosti. Podobně jako u knih bychom také rádi měli možnost dozvědět se základní informace bez nutnosti procházet celý dokument – tedy mít k dispozici obdobu bibliografického záznamu (informace o titulu, jménu autora a případně shrnutí obsahu). Požadavků ze strany uživatelů je celá řada, navíc také existují požadavky tvůrců na snadnost procesu tvorby knihoven, snadnost přidávání nových dokumentů, snadnost omezení přístupu ke konkrétním dokumentům ... Vše je dále ovlivněno vnějšími faktory jako jsou možnost přístupu ke knihovně, právní, sociální a politická omezení a podobně. Digitální knihovny jsou oblastí střetů různých požadavků na zpřístupňování informací řízených dalšími vnějšími vlivy. Snad právě kvůli komplexnosti celé problematiky neexistuje jejich jasné vymezení. Tato práce se zabývá systémem Greenstone a proto bude pojem digitální knihovna chápán podle obecně přijímané definice formulované jeho autory jako:

„specializovaná sbírka digitálních objektů, zahrnujících text, video a audio, spolu s metodami pro přístup k nim a jejich získávání a pro výběr, organizaci a správu sbírky samotné.“ ([44], str. 6)

Pod poněkud vágním pojmem *digitální objekt* si můžeme představit jakoukoliv digitálně uloženou informaci nebo skupinu informací, která má význam pro uživatele knihovny. Jedná se tedy například o textové soubory, obrázky, hudbu nebo animace.

V souvislosti s definicemi digitálních knihoven bývá často diskutována otázka, netvoří-li materiály vystavené na síti internet digitální knihovnu. Jako argument bývá uváděno hlavně množství různorodých materiálů, víc než kolik lze nalézt v jakékoliv knihovně a možnost hledání pomocí internetových vyhledávačů. Je důležité uvědomit si hlavní rozdíl mezi internetem a knihovnou – zatímco digitální knihovna je organizovanou sbírkou digitálních objektů, je hlavním znakem internetu jistá forma anarchie – vystavování dokumentů není řízeno a nikdo neručí za jejich zachování. Organizovanost také přináší pečlivý výběr materiálů zahrnutých do sbírky, což dává jistotu kvalitativně lepších výsledků při hledání informace v knihovně než při prohledávání internetu. Snadno lze nalézt i další rozdíly, ale uvedené odlišnosti již dostatečně demonstrují nestejnorodost obou přístupů.

Při posuzování výsledků dosažených v oblasti digitálních knihoven je nutné mít na paměti, že z hlediska uchovávání informací se jedná o velice mladý obor. První seriózní úvaha na téma digitálních knihoven byla publikována v červenci 1945 vedoucím Úřadu pro vědecký výzkum a vývoj doktorem Vannevarem Bushem. Ústav koordinoval výzkumné úsilí zhruba 6000 vědců v období 2.světové války. Ve svém často citovaném článku *„As we may think“* (viz [7]), se Bush zabýval problémem efektivního shromažďování a rozšiřování informací při intenzivním

vědeckém výzkumu. Pozastavoval se nad tím, kolik výsledků lidského poznání není využito nebo je objevováno znovu jen proto, že neexistuje způsob jak zkušenosti uchovávat a sdílet. V článku se dále věnuje návrhu a popisu zařízení nazvaného *Memex*, určeného na uchovávání a organizaci dokumentů. Ačkoliv řada technických detailů je v dnešní době překonána, některé základní koncepty zůstávají v platnosti stále – například automatické prohledávání rozsáhlých sbírek dokumentů nebo princip hypertextu.

S rozvojem technologií v následujících letech se myšlenka usnadnění zpracování informací s využitím strojů stávala reálnější. V 60. letech 20. století byl v Kongresové knihovně USA (*LoC – Library of Congress*, viz [24]) definován standard MARC (*MAchine-Readable Cataloging*, viz [28]), určený pro výměnu a strojové zpracování katalogových informací mezi knihovnami. Dalším důležitým mezníkem byl vývoj v oblasti osobních počítačů a také rozvoj internetu v poslední dekádě 20. století. Výkonná a dostupná technika spolu s globálním médiem na výměnu informací způsobily velký nárůst digitálních dat a potřebu jejich organizace. Na výzkum v oblasti digitálních knihoven byly zvláště ve Spojených státech vydány značné prostředky, což odráží jejich současnou vedoucí pozici v tomto oboru. Mezi nevýznamnější programy na podporu výzkumu a vývoje digitálních knihoven patří zvláště DLI (*Digital Library Initiative*, viz [11]) z roku 1994 a její nástupce DLI-2 v posledních letech (viz [12], [26]). Digitální knihovny jsou v současné době perspektivním interdisciplinárním oborem s prakticky aplikovatelnými výsledky.

2 Teoretická část

2.1 Pojmy související s digitálními knihovnami

Jak bylo uvedeno v minulé kapitole, digitální knihovny zasahují do celé řady oblastí – od informačních technologií, přes správu informací, komunikaci, právo, uživatelská rozhraní až po knihovnictví, archivaci a manipulaci s cennými historickými artefakty. Popisem využití jednotlivých přístupů a jejich návazností se věnuje řada obsáhlých knih a článků (např. [1], [4], [44]) a pokouší se vymezit pojem digitálních knihoven v rámci komplexního souboru používaných technik. Tato práce je věnována převážně systému Greenstone a její rozsah nedovoluje ani ve stručnosti uvést všechny používané techniky a termíny digitálních knihoven. V rámci této kapitoly jsou shrnuty jen ty nejdůležitější problémy, jejichž znalost je potřebná k základnímu pochopení šíře popisované oblasti.

2.1.1 Dlouhodobé uchovávání informací

Hlavním úkolem každé knihovny je uchovávání informací pro pozdější využití. Stejně tomu je i u knihoven digitálních, které však k uchovávání materiálů používají diametrálně odlišné strategie než jaké se využívají u archivace fyzických artefaktů. V případě knih jsou hlavními faktory čas a prostředí, které znehodnocují nosič informace – knihu – a tím i informaci samotnou. Řešením je vytvoření takového prostředí, které fyzické chátrání knih co nejvíce zpomalí, případně pořizování kopií materiálů, jejichž zničení je nevyhnutelné. Zvláště v poslední době se k převodu chátrajících nebo historicky a kulturně cenných psaných a tištěných materiálů používá digitalizace – tedy převedení do digitální formy, kterou lze uchovávat a zpracovávat pomocí současné výpočetní techniky. Otázkou je, zda tento přístup není příliš krátkozraký. Zatímco technologie výroby a archivace knih počítá v řádech desítek a stovek let, v informačních technologiích se horizont pohybuje v řádech jednotek roků. Nové technologie se objevují a vytlačují ty staré, málokdo používá 10 let staré programové vybavení, kdežto 50 let starou knihu můžeme číst i dnes.

Zvláště u digitálních knihoven získává tento problém na aktuálnosti, neboť jejich úkolem je dlouhodobě uchovávat digitálně uložené informace. V prostředí, v němž není žádná používaná technologie stabilní, nelze stanovit jednotící základ, který by umožnil využívat současné sbírky digitálních dat nejen za 50, ale třeba už jen 10 let. Zastarávání se týká jednak médií, na nichž jsou digitální objekty uloženy (disky, CD-ROM ...), jednak zařízení schopných je číst, ale také nástrojů schopných data správně interpretovat. Uchovávání informací v elektronické podobě přináší velkou výhodu v podobě snadné manipulace, ovšem za cenu značné nejistoty při dlouhodobé archivaci. Strategie používané pro archivaci digitálních materiálů se tedy zaměřují jak na ochranu nosiče informací, tak i na zachování prostředků pro jejich interpretaci.

Ochrana fyzického nosiče se zaměřuje na zabránění ztráty informací v důsledku poškození nosiče, který je obsahuje. Až na předmět zájmu je tato strategie totožná s přístupy používanými u tištěných materiálů. Metodami jsou hlavně kopírování (*replikace*) a obnovování (*refreshing*) uložených dat.

Ochrana informace zahrnuje strategie pro zachování schopnosti uloženou informaci využívat a reaguje tak na neustálou změnu používaných technologií. Jak už bylo zmíněno, zastarávání probíhá jak v oblasti technického vybavení (čtecí zařízení, paměťová média, výpočetní technika), tak i v oblasti programového vybavení. Proto se ochrana informací dále štěpí na dvě strategie. První se zabývá zachováním technického prostředí a to buď přímo udržováním starých

zařízení v chodu (*technology preservation*), nebo jejich simulováním na stávajícím vybavení (*technology emulation*). Ačkoliv je první přístup z dlouhodobého hlediska neudržitelný, bývá alespoň zpočátku méně nákladný než vytvoření simulace. Druhá strategie se zaměřuje na zachování samotné informace v použitelné formě a reaguje tak na stále se měnící formáty souborů. Podobně jako u technického vybavení existují dva přístupy – buď převod dat do nových formátů (*information migration*) nebo jejich zapouzdření do objektů schopných zpřístupňovat jejich obsah (*information encapsulation*).

Protože neexistuje standardizovaný způsob uchovávání digitálních informací, používá většina digitálních knihoven vlastní pravidla a stanovuje formáty dat. Často používaným pravidlem bývá upřednostňování jednoduchých a ověřených formátů a to i za cenu určité ztráty komfortu například při jejich prohlížení.

2.1.2 Digitální a digitalizované materiály

Jednou z charakteristik digitálních dokumentů je způsob jejich vzniku. *Digitální* materiály (v angličtině označované *born digital*) nemají žádný fyzický originál a jsou výsledkem procesu, který produkuje přímo digitální informace. Příkladem může být dokument napsaný v textovém editoru, fotografie pořízená digitálním fotoaparátem nebo skladba zkomponovaná na počítači. Digitální materiály mohou být uchovávány a šířeny v téže kvalitě, v jaké byly vytvořeny.

Digitalizované materiály jsou takové, které vznikly procesem převodu existujících fyzických materiálů do digitální podoby. Nejznámějším příkladem je snímání textu či obrázků (*scanování*). Digitalizace může sloužit pro zpřístupnění materiálů nebo pro jejich archivaci (s ohledem na nejasnosti zmíněné v podkapitole 2.1.1). Proces digitalizace však vnáší do získaného dokumentu šumy a nepřesnosti vzniklé převodem z jednoho média na jiné. Digitalizovaný materiál tedy nikdy nebude ekvivalentní originálu. Přesto ale tyto materiály tvoří důležitou součást velkého počtu sbírek. Často zpřístupňují širokému okruhu uživatelů historicky cenné dokumenty, k jejichž originálu má přístup pouze několik povolaných. V této souvislosti je možné zmínit například projekt *American Memory* Kongresové knihovny (viz [25]), který se zabývá vytvářením sbírek materiálů z historie USA. Cílem těchto sbírek je šířit kulturní dědictví národa a nahradit pouhou ochranu originálů sdílením jejich obsahu, což je jednou z hlavních funkcí každé knihovny.

V souvislosti s digitalizací je třeba alespoň krátce zmínit proces zvaný *OCR* (zkratka z anglického *Optical Character Recognition* – optického rozpoznávání znaků), který slouží pro extrakci textu z obrázku získaného při scanování. Ačkoliv digitální knihovny mohou uchovávat i jen pouhé obrázky stránek textu, je možnost získání zobrazeného textu důležitá pro poskytování služeb jako je například fulltextové vyhledávání. Uživatel pak může zadat hledané slovo vyskytující se kdekoliv v textu a digitální knihovna mu nabídne buď obrázek původní stránky, nebo její text, případně kombinaci obojího.

2.1.3 Přístup k informacím

Kromě uchovávání informací je úkolem většiny knihoven také jejich zpřístupňování. Velká sbírka pečlivě vybraných a udržovaných dokumentů ztrácí svůj účel, není-li kdo by ji využíval. U klasických knihoven může tento požadavek vyvolat rozpor s posláním informace uchovávat, protože půjčováním fyzických kopií děl se tyto postupně znehodnocují. Zde se projevuje jedna z výhod digitálních knihoven – uchovávané materiály mohou být nabídnuty neomezenému počtu zájemců aniž by došlo k jejich poškození. Proto, neexistují-li právní omezení, není žádné rozdělení na přístupné a nepřístupné materiály z hlediska nutnosti zachovat je v původním

stavu. Digitalizovaná kopie Guttenbergovy bible může být nabídnuta k prohlédnutí komukoliv, kdežto k originálu má přístup pouze omezený počet lidí.

Druhým aspektem přístupu k informacím je forma, jakou jsou uživateli předkládány. Případný zájemce o Guttenbergovu bibli nemusí stát o znění textu, ale chtěl by vidět, jak byla vytištěna. Lingvista by naopak dával přednost co nejpřesnějšímu znění textu a možnosti v něm vyhledávat. Digitální knihovny mohou nabídnout obojí a umožnit tak uspokojit různorodé požadavky. Navíc dobrá prezentace obsahu může tvořit přidanou hodnotu, která z knihovny dělá užitečný nástroj. I kdyby měl například zmíněný lingvista k dispozici originální výtisk, nebyl by v něm schopen vyhledávat tak rychle, jak mu to umožňuje digitální knihovna. Na druhou stranu sebelepší služby nenahradí pocit ze čtení originálu a obracení stránek.

2.1.4 Jednoznačná identifikace dokumentů

V elektronickém světě propojeném globální sítí ztrácí fyzické umístění význam a pro lokalizaci a ověření pravosti dokumentu je třeba mít mechanismus, který by každý dokument jednoznačně identifikoval. Paralelou v reálném světe může být například mezinárodní identifikátor knih *ISBN*. Ve světe elektronických dokumentů však bohužel žádný takový celosvětově jednotný standard neexistuje a proto se požadavky na jednoznačnou identifikaci zatím nesetkávají s úspěchem. Většina digitálních knihoven se drží jednoznačné identifikace alespoň v rámci svých sbírek, což je důležitým předpokladem jejich správného fungování.

Identifikace je úzce propojena s tématy diskutovanými v této kapitole. Bez jednoznačné identifikace není možné zajistit kvalitní vyhledávání dokumentů ve sbírce, ověřovat jeho pravost nebo zajišťovat přístupová práva. Není také možné dobře prohledávat sbírky udržované různými digitálními knihovnami – na jediný dotaz můžeme získat řadu výsledků, z nichž však část může být odkazem na stejný dokument, pouze označeným jiným identifikátorem a uchovávaný jinou knihovnou. Zmiňme alespoň základní existující přístupy k vytvoření globálního identifikačního schématu:

Nepřímá adresace zavádí prostředníka, který na základě jednoznačného identifikátoru zprostředkovává informace o aktuálním umístění daného zdroje. Nepřímá adresace počítá s globálním přidělováním unikátních identifikátorů, které by existovaly i po zániku zdroje, který označují – jsou tedy přiděleny právě jednou a jsou persistentní. Při potřebě získat dokument s daným identifikátorem je nejprve poslán dotaz zprostředkovateli služeb nepřímé adresace, který na základě identifikátoru určí aktuální umístění dokumentu, případně oznámí jeho zrušení. Koncept nepřímé adresace je vyžíván například v systému *PURL* (*Persistent URL*, viz [33]), *URN* (*Uniform Resource Name*, viz [21]) nebo *Handles* (viz [10]).

Vytváření identifikátoru v závislosti na obsahu (c*ontent-based computations*) – vytváření jednoznačných „otisků“ (*hash*) souborů. Tento způsob garantuje téměř dokonalou jednoznačnost identifikátorů, na druhou stranu je velice citlivý na jakoukoliv změnu obsahu. Sice tak umožňuje uchování integrity dokumentu, ale zároveň každou verzi dokumentu obsahující stejnou informaci může označit jiným identifikátorem například jen kvůli změně kódování. Příkladem může být generování identifikátorů s využitím algoritmu *MD-5* (viz [52]).

Změna používání současných schémat přidáním nové funkčnosti nebo služby. Typickým příkladem je projekt *ARK* (*Archival Resource Key*, viz [8]), který prosazuje využívání současného mechanismu URL s přidáním služeb určujících konkrétní lokaci zdroje. Podobně jako u nepřímé adresace by tedy adresa byla nejprve předána prostředníkovi, který by vyhodnotil aktuální umístění zdroje. ARK ale přenechává zodpovědnost za platnost umístění

zdroje jeho autorovi. Pokud se tedy zdroj přesune, je na autorovi, aby o této změně dal vědět službě řídící převod virtuálních adres na absolutní.

2.1.5 Metadata

"Jestliže dokumenty jsou základními kameny digitální knihovny, pak značkování a metadata jsou jejími základními elementy organizace. Značkování je použito pro určení struktury jednotlivých dokumentů a řízení způsobu, jakým jsou prezentovány uživateli při prohlížení. Metadata jsou používána k urychlení přístupu k relevantním částem sbírky za pomocí hledání a procházení." ([44], str. 221)

Termín metadata se stal s nástupem informačních technologií populární, často se však zapomíná, že metadata jsou již dlouhou dobu využívána. Příkladem mohou být knihovnické katalogy, sloužící pro rychlé vyhledávání konkrétních děl v rámci knihovny. Jedna z obecných definic označuje metadata jako „data o datech", jedná se tedy o údaje, které většinou slouží k popisu informačních zdrojů. Zvláště pro (digitální) knihovny poskytují velice důležitá vodítka pro organizaci dokumentů, jejich vyhledávaní a vytváření souhrnů sloužících uživatelům k získání představy o obsahu konkrétního dokumentu bez nutnosti jeho procházení.

Metadata lze podle oblasti použití dále dělit do různých kategorií. *Značkovací* (*markup*) metadata slouží pro určení struktury dokumentu, nebo pro definici formátování (v takovém případě je ale bereme spíše jako součást dokumentu) – příkladem značkovacího jazyka je HTML (viz [41]) určený pro tvorbu internetových stránek. *Metadata pro popis zdrojů* (*resource describing*) popisují vlastnosti informačních zdrojů – autora, datum vydání, obsah a podobně a jsou velice důležitou součástí každé spravované sbírky. Existují různé formáty pro zápis metadat, nejdůležitější však jsou *SGML* (*Standard Generalized Markup Language*, viz [42]) a XML (*eXtensible Markup Language*, viz [23], [43]). Zatímco SGML je velice komplexní jazyk určený spíše pro popis značkovacích formátů než pro značkování samotné a je většinou používán pouze velkými institucemi, je XML jeho zjednodušenou verzí určenou speciálně pro podporu interoperability na síti. XML poskytuje dobrý způsob jak charakterizovat strukturu dokumentu a metadata a proto je vhodný pro použití v digitálních knihovnách. Poslední skupinou metadat jsou *administrativní a práva spravující* metadata která slouží pro definování pravidel manipulace s dokumenty a zachování jejich integrity.

Z hlediska způsobu vzniku pak rozlišujeme mezi metadaty *explicitními* a *extrahovanými*. *Extrahovaná* metadata se získávají automatizovanými procesy z obsahu dokumentu samotného. Při těchto postupech se uplatňují například techniky *dolování textu* (*text mining*) nebo *automatická extrakce zkratek a frází*. Část informací s sebou nese také formát, v němž jsou data uložena – z přípon souborů lze určovat typy, z hlaviček grafických souborů například informace o rozlišení nebo barevné hloubce. Kvalita extrahovaných dat se však může velmi lišit a zvláště u souborů s ne přesně stanoveným formátem může poskytovat pouze neúplné informace.

Naproti tomu *explicitní* metadata jsou většinou vytvářena a přiřazována odborníkem po důkladném studiu dokumentu. Tento přístup poskytuje velice kvalitní informace o zdrojích. Jsou-li použita jako základ sbírky, umožňují dobré vyhledávání a popis materiálů. Oproti extrahovaným metadatům jsou však mnohonásobně dražší a jejich tvorba se tak může stát významnou položkou v celkové ceně zařazení dokumentu do sbírky. Uvádí se, že například vytvoření bibliografického záznamu pro systém MARC zabere zkušenému knihovníkovi mezi jednou až dvěma hodinami práce.

Způsob jakým budou jednotlivé informační zdroje popsány, tedy pravidla pro zápis metadat, se nazývá *metadatové schéma*. Primárně se schémata odlišují podle oblasti nasazení a

odpovídají tak rozdělení metadat popsanému v předchozím textu. I v rámci jedné oblasti nasazení však mohou existovat různá schémata a to buď proto, že jsou využívána různými zájmovými skupinami (metadata pro fyziky budou mít jiné prvky než metadata pro literární vědu) nebo z důvodů neshodnutí se na standardu. Na závěr této kapitoly jsou uvedeny dva rozdílné přístupy využití metadatových schémat.

Již zmiňovaný MARC (*MAchine-Readable Cataloging*, viz [28]) je metadatové schéma určené pro automatické zpracování a výměnu dat mezi knihovnami. Představuje takzvaný maximalistický přístup, protože jeho definice se snaží postihnout každý potenciálně potřebný aspekt dokumentovaného zdroje zajímavý z hlediska knihovnictví. Záznam pro systém MARC se skládá ze stovek různých informací členěných do polí a podpolí. K jeho vytvoření je třeba nezanedbatelného množství času a úsilí zkušených pracovníků. Výsledkem je ale velice kvalitní popis informačního zdroje.

Druhým, takzvaně *minimalistickým,* přístupem je schéma *Dublin Core* (většinou zkracováno na *DC*, viz [14]). Na rozdíl od schématu MARC definuje jen 15 základních prvků s možností jejich rozšíření. Hlavním cílem DC je umožnit všem tvůrcům jakýchkoliv typů dokumentů jednoduše popsat své výtvory a umožnit tak jednak zlepšení stávajících služeb vyhledávání na síti internet, jednak realizaci nových technologií pro kvalitní správu dokumentů (viz například OAI v kapitole 2.1.6).

2.1.6 Interoperabilita

"Základním problémem digitálních knihoven je interoperabilita: schopnost výměny a sdílení dokumentů, dotazů a služeb. Interoperabilita se také týká výměny zmíněného mezi různými komponentami jedné digitální knihovny. Z trochu jiné perspektivy je interoperabilita v digitálních knihovnách schopností vytvářet jediný (virtuální) náhled na mnoho různých komponent bez obětování jejich autonomie." ([15])

Interoperabilita, jeden z nejdůležitějších požadavků kladených na digitální knihovny, je sama o sobě natolik širokým tématem, že není možné ve zkratce vyjádřit všechny její aspekty. Zabývá se výměnou informací mezi různými systémy za použití různých komunikačních kanálů a různých metod přenosu dat. Interoperabilita má umožnit zapojit jednotlivé autonomní systémy do spolupracující sítě, která bude uživateli poskytovat služby jako jednotný celek. Klasické knihovny za dlouhou dobu své existence vyvinuly fungující systém vzájemných (meziknihovních) výpůjček a sdílení knihovnických záznamů a dosáhly tak dobré úrovně interoperability. U digitálních knihoven komplikuje zavedení podobného systému hlavně různorodý technologický základ použitý pro tvorbu a fungování různých systémů. Každá digitální knihovna může uchovávat (a zpravidla i uchovává) informace o spravovaných dokumentech ve vlastním formátu. Navíc různé digitální knihovny fungují na různých platformách a neexistují všemi používané standardy pro výměnu informací. Na vyřešení těchto i dalších problémů se stále pracuje. V následujících odstavcích budou stručně popsány dva přístupy k interoperabilitě, které podobně jako u metadatových schémat v kapitole o metadatech představují maximalistický a minimalistický přístup k řešení problému.

Příkladem maximalistického přístupu je protokol *Z39.50* (viz [27]) určený pro získávání informací mezi klientem a databázovým serverem. *„Z39.50 je příkladem aplikační vrstvy referenčního modelu OSI (Open System Interconnection), úplného standardu pro prostředí počítačových sítí."* ([44] str. 247) Tento protokol je velice rozsáhlý a umožňuje výměnu informací (dokumentů i metadat) mezi různými systémy. Není zaměřen jen na použití v knihovnách, ale podporuje také komunikaci mezi muzei, galeriemi a podobnými institucemi.

Standard je natolik obsáhlý, že obvykle nebývá implementován v informačních systémech celý. Jeho jedenáct částí pokrývá komunikaci mezi klientem a serverem od zahájení spojení (jedná se o stavový protokol) přes vyhledávání a získávání materiálů, práci se seznamem výsledků, řízením přístupu a procházením materiálů až k správnému ukončení spojení. Navíc nabízí rozšířené služby, které jdou nad rámec základně nabízených. Z těchto 11 částí je po každém systému, který hodlá protokol využívat, požadována implementace pouze 4 oblastí – *inicializační část*, *vyhledávací služby*, *prezentační služby* a *dotazy prvního typu*. Výhodou protokolu Z39.50 je jeho velká obecnost, která umožňuje výměnu informací i mezi značně odlišnými systémy, jeho nevýhodou pak je značná složitost a nároky na implementaci.

Druhý přístup odděluje poskytovatele služeb (například vyhledávání) od poskytovatelů dat. Minimalistickým se dá nazvat proto, že vychází vstříc tvůrcům dokumentů a jedinou zodpovědnost, kterou jim dává, je vystavení metadat (výchozím schématem je obvykle Dublin Core) a zprovoznění jednoduché služby pro jejich zpřístupnění. Specializované nástroje nazývané *sběrače metadat* (*metadata harvesters*) pak procházejí vystavená metadata a ukládají je do databází. Poskytovatel služeb s využitím takto získaných metadat může umožnit hledání a určování umístění dokumentů. Příkladem využití tohoto principu je přístup *OAI* (*Open Archives Initiative*, viz [34]).

2.1.7 Právní, politický a sociální aspekt

Většina knih a článků o digitálních knihovnách nezapomene zmínit problémy, které se z pohledu tvůrců knihoven zaměřených spíše na oblast informačních technologií nemusejí zpočátku zdát důležité. Systematická organizace dokumentů a jejich vystavování, zvláště pokud se jedná o prostředí světové sítě internet, s sebou nese závažné sociálně-právní aspekty. Zjednodušující přístup uživatele internetu, že „všechno patří všem“ a „co je dostupné na síti lze také bez omezení použít,“ je nesprávný. V případě digitálních knihoven je ale přehlížení právních otázek, zejména z oblasti práva autorského, naprosto nepřípustné (důsledkem by bylo odmítnutí digitálních knihoven v praxi ze strany autorů a vlastníků autorských práv). Opomineme-li materiály vytvořené pro vlastní potřebu, které však u významných sbírek tvoří pouhý zlomek obsahu, je důležité před zařazením libovolného materiálu do sbírky zajistit, že se tak děje se souhlasem autora, případně s držiteli autorských práv. Na rozdíl od děl majících fyzickou podobu se elektronický dokument stává okamžikem svého vystavení na síti dostupný komukoliv z celého světa. Znění autorského práva se ve většině zemí liší a proto je nutné mít na paměti i tuto skutečnost. Digitální knihovna by také měla zajišťovat mechanismy umožňující částečné nebo i úplné omezení přístupu k jednotlivým dokumentům nebo dokonce jejich odstranění ze sbírky, nastane-li taková potřeba.

I obsah dokumentů se může stát předmětem sporu. Díky dostupnosti z celého světa se ke sbírkám mohou dostat lidé z různých kulturních a sociálních prostředí, kteří mají různé názory na to, co je přijatelné a co by za žádných okolností nemělo být zveřejňováno. Neopatrná manipulace s takto citlivými dokumenty může vyvolat spory nejen v rovině tvůrce sbírky – uživatel, ale může přerůst i v mezinárodní skandál. Množství omezení kladených na obsah a způsob prezentace sbírek digitálních knihoven je velké a není v silách jednoho člověka všechny obsáhnout. Jak již bylo několikrát uvedeno, digitální knihovny jsou natolik provázány s různými a při prvním pohledu nesouvisejícími oblastmi lidského života, že jejich tvorba a správa by měla být výsledkem společného úsilí odborníků z různých oblasti. Jen tak lze zajistit, že budou když ne všechny, tak alespoň hlavní aspekty vystavování materiálů zváženy a díky tomu bude možné se v budoucnu vyhnout případným sporům.

2.2 Digitální knihovna Greenstone

Tato práce je věnována zejména existujícímu systému na tvorbu digitálních knihoven s názvem Greenstone (*Greenstone Digital Library software*, dále bude označován jen jako Greenstone – viz [20]). Jedná se o volně dostupný soubor programových nástrojů umožňujících vytváření a provoz digitálních knihoven, který nabízí univerzálně použitelné řešení odpovídající požadavkům popsaným v minulé kapitole.

Greenstone vznikl v roce 1995 a je vyvíjen v rámci projektu *NZDL* (*New Zealand Digital Library*, viz [31]) na univerzitě Waikato na Novém Zélandu. Na jeho tvorbě se podílí řada lidí převážně z univerzity Waikato, hlavními návrháři a vývojáři jsou Rodger McNab a Stefan Boddie, kteří využívají poznatků formulovaných profesorem Ianem H. Wittenem. Ačkoliv na počátku stála myšlenka vytvoření snadno přístupného archivu vědeckých článků s možností vyhledávání, postupem času se Greenstone vyvinul v plnohodnotný software na správu digitálních knihoven. Na neustále rostoucím projektu začal pracovat velký tým odborníků z různých oblastí: počítačová grafika, počítačem podporovaná spolupráce, interakce člověka s počítačem, zpracování obrazu, knihovní věda, multimédia, etnografie, strojové učení, muzikologie a analýza a návrh systémů. Práce na projektu poskytuje rámec pro výzkum v různých oblastech, jehož výsledky jsou začleňovány do systému Greenstone. Ten je v současné době používán několika agenturami Organizace Spojených národů jako jsou například FAO (*Food and Agriculture Organization*) v Římě, UNESCO v Paříži, Univerzita Spojených Národů v Tokiu nebo Centrum pro lidské osídlení (*Habitat*) v Nairobi. Speciálně vytvořené sbírky jsou distribuovány na médiích CD-ROM v rozvojových zemích a umožňují překonávat problémy při zavádění nových technologií a pomáhají zvyšování životní úrovně.

Digitální knihovny vytvořené za pomocí systému Greenstone poskytují vhodný způsob organizování informací a jejich zpřístupňování na internetu. Informační sbírky kombinují prostředky pro rozsáhlé vyhledávání v textu (*fulltextové vyhledávání*) založené na různých typech metadat. Uživateli jsou nabízeny různé možnosti přístupu ke spravovaným informacím, ačkoliv jejich šíře se liší sbírku od sbírky podle použitých metadat a rozhodnutí tvůrce. Aby vyhověl neobvykle širokým požadavkům kladeným na digitální knihovny, je Greenstone vyvíjen pod *GNU* licencí v duchu *open-source* software. Uživatelé jsou motivování k úpravám a vylepšením systému a tím pomáhají rychlejšímu rozvoji a rychlé odezvě na neustále se objevující nové požadavky. Pro konzultace problémů a diskuze ovlivňující další vývoj celého systému byly zřízeny konference jak pro uživatele, tak i pro tvůrce a správce sbírek. Tímto způsobem a zpřístupněním archivů zpráv je možné rychle získat odpovědi na důležité otázky a konzultovat problémy nejen s ostatními uživateli, ale i se samotnými tvůrci systému Greenstone. Velká obecnost, která stojí v základech návrhu, umožňuje přizpůsobit podobu a chování systému různým jazykovým i kulturním zvyklostem. Kromě angličtiny existují kompletní překlady uživatelského rozhraní také do španělštiny, ruštiny a francouzštiny. České rozhraní bylo pro dřívější verze vytvořeno panem Romanem Chýlou, v rámci této práce byl dokončen překlad pro verze nové. Pro další jazyky existují překlady důležitých součástí rozhraní a zájemci mohou pomoci v dalším překládání.

2.2.1 Knihovny, sbírky a dokumenty

Strukturu organizace materiálů vytvářenou systémem Greenstone lze rozdělit do jednoduché hierarchie: knihovny, sbírky, dokumenty. *Dokumenty* jsou základními stavebními jednotkami nejnižší úrovně, obsahující samotné informace, o jejichž zpřístupňování se knihovna stará. Mohou obecně obsahovat text, ale také obrázky, zvukové nahrávky nebo video. Jejich formát

není téměř omezen, neboť začleňování nových dokumentů do sbírek je řízeno a ovlivňováno systémem speciálních modulů nazývaných *pluginy*, které lze relativně snadno upravit nebo vytvořit nové. Díky tomu je možné vytvářet sbírky obsahující nejen dokumenty známých formátů, ale definovat i postup při zpracování dokumentů do té doby knihovně neznámých. Moduly také určují jakým způsobem budou k danému dokumentu získána metadata – může být proveden pokus o jejich extrahování z obsahu, převzetí metadat přiřazených externě nebo kombinace obojího.

Sbírky zastupují větší množství dokumentů rozličných formátů sdružených do jednoho organizačního celku. Jejich velikost se může lišit od několika desítek až po stovky tisíc či jednotky miliónů dokumentů. Lze je přirovnat k oddělením v klasických knihovnách – v každé sbírce většinou najdeme tematicky příbuzné materiály, ačkoliv rozhodnutí o obsahu sbírky záleží pouze na tvůrci. Příkladem mohou být sbírka dokumentů o populárním filmu, fotografie z dovolené nebo třeba rady pro chování šneků. Sbírky zároveň poskytují jednotné uživatelské rozhraní umožňující využívat služeb digitální knihovny – nabízí různé náhledy na dokumenty, zpřístupňují vyhledávání a dodatečné organizační struktury pro lepší orientaci. Většina digitálních knihoven v souladu s analogií s knihovnami klasickými obsahuje větší množství sbírek.

Knihovny obecně obsahují mnoho různých sbírek, z nichž každá může být organizována jinak. Většinou však zůstává zachována podobnost při prezentaci obsahu těchto sbírek uživateli (jednotná prezentace přispívá k přehlednosti a snazší práci s digitální knihovnou). Při zjednodušeném náhledu je knihovna souborem nástrojů a struktur zpřístupňujících jednotlivé sbírky a dokumenty v nich, které poskytují prostředky na zobrazování, procházení a vyhledávání. Při bližším zkoumání různých scénářů nasazení digitálních knihoven, zvláště v prostředí sítě internet, se pak spíše než o programy a procesy jedná o komplexní systém nabízející služby kvalitativně vyšší úrovně. Zvláště návrh systému Greenstone verze 3 (viz [13]) slibuje do budoucna větší možnost distribuovaného vytváření a správy sbírek.

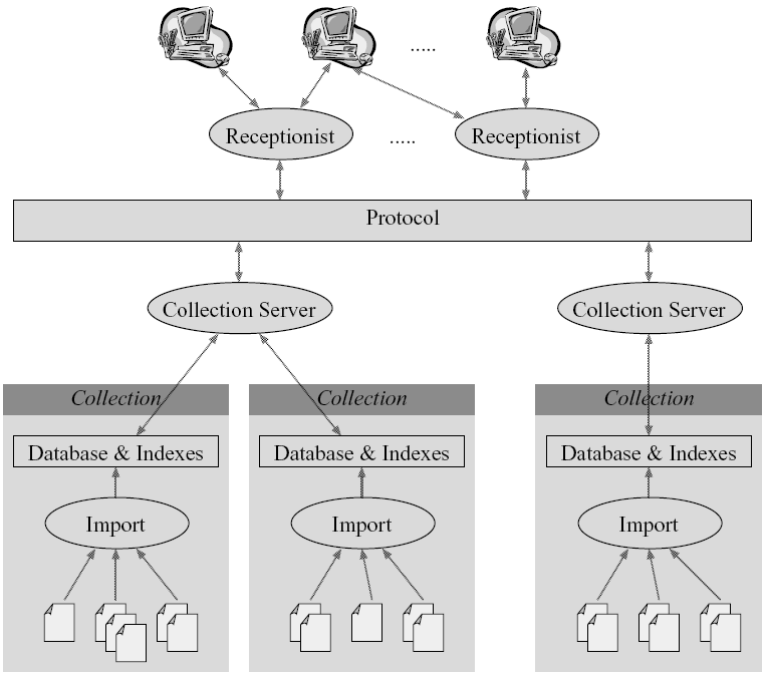
2.2.2 Obecná struktura systému Greenstone

V této i několika následujících kapitolách jsou popsány důležité části systému Greenstone, které se podílí na tvorbě jednotlivých sbírek i fungování celé knihovny. Informace jsou převzaty z literatury (viz [3], [20], [44], [45], [46], [47], [48], [49]) a doplněny o vlastní poznatky získané při práci s digitální knihovnou Greenstone.

Jádro systému je napsáno v programovacím jazyce C++ (viz např. [40]) a ve velké míře využívá mechanismu virtuální dědičnosti. Zdrojové kódy jsou k dispozici nejen pro prohlížení, ale je možné je také upravovat a překladem vytvářet vlastní verze systému. Díky velké míře obecnosti použité při návrhu i implementaci a zvláště díky použití externích modulů napsaných v jazyce Perl (viz např. [37]) lze ale do značné míry ovlivnit procesy využívané při budování sbírek bez nutnosti do jádra zasahovat. Rozdělení jednotlivých procesů do vrstev komunikujících přes jasně stanovená rozhraní také umožňuje upravovat chování jen určitých částí při zachování funkčnosti ostatních. Například prezentace obsahu digitální knihovny, tedy jednotlivých sbírek obsahujících dokumenty, je vytvářena až za běhu na základě informací získaných zpracováním dotazu jádrem systému. Ve standardním případě se jedná o vytvoření internetových HTML stránek a jejich zobrazení v prohlížeči. V případě potřeby je ale možné tuto vrstvu zcela nahradit například rozhraním implementovaným v programovacím jazyce Java (viz [38]) a zcela tak změnit vzhled a způsob přístupu ke knihovně. To vše bez nutnosti zasahovat do částí systému, které obstarávají správu a vyhledávání ve sbírkách samotných. Díky

rozdělení na vrstvy se stává struktura digitální knihovny pro uživatele transparentnější – jednotné uživatelské rozhraní může poskytovat informace získané ze sbírek spravovaných různými digitálními knihovnami po celém světě.

Na následujícím obrázku (převzatém z [3], str. 56) je zobrazeno několik uživatelů (zastoupených ikonami počítačových terminálů) přistupujících ke třem různým sbírkám (jsou zobrazeny v dolní části). Před tím, než jsou sbírky dány k dispozici, musejí nejprve projít procesy importu a budování, které zpracují výchozí dokumenty a na základě získaných informací vytvoří indexy a další struktury umožňující prohledávání a procházení sbírek. Oba tyto procesy budou podrobněji popsány v následujících kapitolách.



Obrázek 1: Obecná struktura systému Greenstone

V návrhu systému hrají důležitou roli dva klíčové procesy: „recepční“ (*receptionist*) a „server sbírky“ (*collection server*). Z pohledu uživatele poskytuje recepční přístup k digitální knihovně a ukrývá před ním detailní strukturu celého systému. Recepční se stará o zpracování vstupů zadaných uživatelem, v typickém případě zadávaných pomocí klávesnice nebo myši přes prohlížeč internetových stránek (*webovský prohlížeč*). Po analýze příkazů pak odešle požadavek příslušnému serveru sbírky (obecně jich může být více). Ten při řešení dotazů využívá datové struktury vytvořené při budování sbírky pro vyhledávání patřičných dokumentů nebo sbírek. Výsledek hledání je předán zpět recepčnímu, který jej zobrazí uživateli. Servery sbírek tedy poskytují abstraktní mechanismus pro manipulaci s obsahem sbírek, recepční jsou zodpovědní za uživatelské rozhraní. Zvláště v případě přístupu k digitální knihovně pomocí internetových stránek sdílí více uživatelů téhož recepčního. Ve většině případů také různé sbírky sdílí jeden server sbírek. Popsaná architektura je velice přizpůsobivá a umožňuje také, aby jeden recepční komunikoval s více servery sbírek a naplňuje tak scénář popsaný v úvodu této kapitoly.

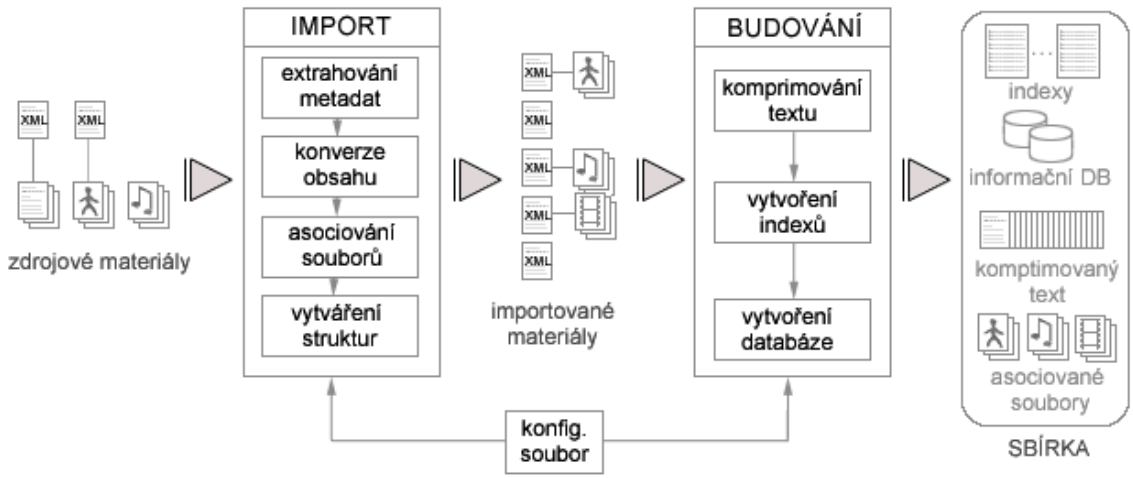
Procesy typu „recepční“ komunikují s procesy typu „server sbírek“ prostřednictvím jednotného *protokolu*. Implementace tohoto protokolu závisí na způsobu použití digitální knihovny Greenstone. V prostředí distribuovaných systémů s nezávislými procesory, ve kterém každý server sbírky může fungovat na jiném stroji, se jedná o mechanismus zabezpečující

spolupráci všech součástí. V současné době nejběžnějším a také nejjednodušším případem použití je scénář, při kterém oba procesy, recepční i server sbírek, běží na jednom stroji. Takové prostředí redukuje protokol na pouhé volání funkcí – ačkoliv veškerá komunikace mezi oběma moduly stále ještě probíhá přes rozhraní protokolu. Tato implementace se nazývá *null protocol* a slouží ke zefektivnění a urychlení zpracování odezev na reakce uživatele. Oba moduly jsou v tomto případě zahrnuty v jednom programu, který tvoří fungující jádro celé knihovny. Standardní instalace systému Greenstone obsahuje právě tento způsob distribuce jádra.

Stejný protokol je také implementován s použitím architektury *CORBA* (*Common Object Request Broker Architecture*, viz [32]), která umožňuje vývoj sofistikovanějších uživatelských rozhraní (viz [6]). CORBA používá sjednocené objektově orientované paradigma pro umožnění sdílení objektů procesy fungujícími na různých platformách a implementovaných v různých programovacích jazycích. Právě využití takového protokolu umožňuje realizaci digitální knihovny v prostředí distribuovaných výpočetních systémů.

2.2.3 Tvorba sbírek

Nejdůležitějšími součástmi každé digitální knihovny systému Greenstone jsou sbírky. Při jejich vytváření se uplatňuje jedna z velkých výhod využití výpočetní techniky a tou je možnost automatizace rutinních a časově náročných operací. V případě knihoven je touto operací hlavně vytváření metadat sloužících pro efektivní hledání v rámci sbírek a organizaci obsaženého materiálu – tedy tvorba rejstříků a dalších pomocných struktur. Digitální knihovna Greenstone snímá veškerou odpovědnost za tento proces z tvůrce a provádí jej zcela automatizovaně. Po získání potřebných metadat jsou vytvářeny pomocné struktury určené pro rychlé vyhledávání (indexy), obsahy dokumentů jsou komprimovány pro úsporu místa a výsledný kompaktní celek je připraven pro zařazení do digitální knihovny.



Obrázek 2: Schéma tvorby sbírky

Celý proces tvorby materiálů ze zdrojových souborů různých formátů je rozdělen na dvě části: *import* a *budování* (*building*), které jsou popsány v následujících podkapitolách. Ačkoliv naprostá většina operací prováděných v rámci vytváření sbírek probíhá zcela automaticky, neznamená to, že by tvůrce neměl možnost do procesu zasahovat. Kromě parametrů určujících základní vlastnosti procesů samotných jsou požadavky na podobu a strukturu výsledné sbírky detailně specifikovány v takzvaném *konfiguračním souboru sbírky*. V něm jsou určeny druhy

indexů, které mají být vybudovány, formáty souborů a nastavení způsobu jejich začlenění, nabízené způsoby procházení obsahu sbírky a také způsob prezentace uživateli. Některá z nastavení hrají roli při importu a budování, jiná se projevují až při prohlížení výsledné sbírky a lze je u existujících sbírek měnit a ovlivňovat tak jejich podobu. Rozdělení tvorby sbírky na dva samostatné kroky má své opodstatnění v úspoře času při převodu dokumentů do interního formátu Greenstone, extrakci metadat a generování identifikátoru. V případě přidávání nových dokumentů do již existující sbírky totiž stačí výše uvedené kroky, které jsou součástí importu, provést pouze pro nové materiály. Vybudování sbírky v nové podobě sice vyžaduje její kompletní přestavbu (tedy spuštění budování sbírky), ale i tak u sbírek obsahujících větší množství dokumentů ušetříme čas.

V následujících podkapitolách jsou stručně popsány důležité kroky při vytváření sbírek a jejich význam pro digitální knihovnu v systému Greenstone. Na závěr budou popsány dva nástroje sloužící pro usnadnění tvorby sbírek.

2.2.3.1 Importování materiálů

Hlavním úkolem ve fázi importu je převedení dokumentů z jejich původních formátů do interního formátu systému Greenstone, který je dále využit pro vytváření indexů a struktur tvořících sbírku samotnou. Zároveň je provedeno přiřazení metadat a jsou vytvořeny dodatečné soubory řídící navazující proces budování sbírky.

Jednou z velkých výhod digitální knihovny Greenstone je schopnost zpracovávat a uchovávat soubory téměř jakýchkoliv formátů. Tuto důležitou vlastnost, která reaguje na velké množství existujících formátů a jejich neustálé změny, umožňuje systém modulů nazývaných *pluginy* (viz [46]). Jedná se bloky kódu napsané v jazyce Perl a distribuované jako součást instalace. Z hlediska jádra systému, popsaného v minulé kapitole, se tedy jedná o externí součásti. Díky použitému jazyku je jejich úprava, případně tvorba nových, pro zkušeného uživatele relativně snadná a nevyžaduje žádné zásahy do jádra digitální knihovny ani kompilování. Moduly pro nejznámější formáty (`txt`, `html`, `pdf`, `jpeg`, `doc` a další) jsou k dispozici. Díky využití dědičnosti lze vytvářet nové pluginy pouhým předefinováním některých funkcí a není nutné vytvářet kód celý. Popis tvorby a fungování pluginů je součásti praktické části této práce (viz kapitola 3.5).

Většina zpracování ve fázi importu je řízena právě popsanými moduly. Soubory uložené v adresáři určenému pro import jsou postupně předávány pluginům specifikovaným v konfiguračním souboru sbírky. Modul, který je schopen daný soubor zpracovat pak většinou provede následující kroky (ne nutně v uvedeném pořadí):

- Vytvoří objekt, který bude zastupovat dokument v rámci sbírky.
- Vygeneruje unikátní identifikátor (*OID – Object IDentifier*), kterým bude dokument reprezentován v rámci sbírky a přiřadí jej zástupnému objektu.
- Extrahuje z obsahu zdrojového dokumentu metadata (je-li to možné) a spolu s metadaty definovanými externě je přiřadí zástupnému objektu.
- Zpracuje samotný obsah zdrojového dokumentu. Nejprve jej převede do kódování UTF-8 a poté do formátu vhodného pro zobrazení v prohlížeči HTML stránek. Výsledek je uložen do zástupného objektu.
- Připojí soubory (například obrázky) k zástupnému objektu.
- Uloží zástupný objekt do XML souboru, který dále slouží ve fázi budování sbírky.

Výsledkem importu je soubor obsahující všechny důležité informace o dokumentu i jeho obsahu spolu s relevantními přílohami (obrázky, dokumentem v původním formátu, ...) uloženými ve zvláštním adresáři. Veškeré údaje zpracované při importu jsou pak dále využity při budování sbírek.

Důležitým bodem, který je potřeba zdůraznit, je generování unikátního identifikátoru (*OID*). Jak bylo zmíněno v přehledu důležitých pojmů z oblasti digitálních knihoven, identifikace dokumentů v rámci sbírek a knihoven je klíčová pro jejich správné fungování (viz kapitola 2.1.4). Greenstone k identifikaci dokumentů využívá unikátních řetězců vytvářených v závislosti na obsahu souboru (takzvané *hashování*). Při generování těchto řetězců není využívána žádné standardní metoda (jako je například MD-5), ale algoritmus vytvořený jedním z autorů systému. Díky generování identifikátorů na základě obsahů souborů je možné zajistit téměř dokonalou identifikaci a v případě duplicitních materiálů i ušetření nadbytečných dat. Pokud je například vytvářena sbírka, v níž je stejný dokument uložen několikrát na různých místech, bude do výsledného celku zařazen jen jeden jeho výskyt, protože ostatní mají totožný otisk (*hash*). Přitom ve sbírce může být stále odkazován na různých místech, rozdíl je pouze v úspoře prostoru na disku a nezaplnění indexových struktur zbytečnými záznamy. Předdefinovaný způsob generování identifikátorů v systému Greenstone je možné změnit a definovat vlastní způsob. Této možnosti bylo využito při vytváření ukázkové sbírky DKF, jak je popsáno v příslušné kapitole praktické části (viz kapitola 3.2.3).

2.2.3.2 Budování sbírky

Ve fázi budování sbírky jsou na základě informací uložených v rámci objektů získaných ve fázi importu vytvářeny všechny struktury potřebné pro fungování sbírky. Stejně jako u předchozího kroku je i budování sbírky řízeno nastaveními konfiguračního souboru. Zejména se jedná o počet a druh indexů a klasifikátorů. Proces potřebuje několik průchodů připravenými dokumenty k tomu, aby vytvořil kompletní sbírku. První dva průchody jsou třeba na zkomprimování textů dokumentů. Pro každý vytvářený index jsou provedeny dva další průchody. Na závěr celého procesu je vytvořena hlavní databáze (*informační databáze*), uchovávající veškeré důležité informace o sbírce. Výsledkem jsou soubory obsahující indexy, komprimované texty, informační databázi a dále soubory připojené k původním dokumentům (například již zmiňované obrázky).

Vytvářené indexy zpřístupňují takzvané fulltextové prohledávání. Jejich struktura umožňuje efektivní vyhodnocování dotazů na výskyt libovolného slova v rámci textů všech dokumentů a vybraných metadat obsažených ve sbírce. Indexy je možné vytvářet v různých úrovních podrobnosti – od celých dokumentů přes jednotlivé kapitoly či odstavce až po samotný text. V rámci zefektivnění prohledávání se také využívají postupy na potlačení významu velikosti písma (*case-folding*) a vyhledávání podle kořenů slov (*stemming*). Poslední uvedená technika nejprve u všech slov dotazu nalezne příslušné kořeny a poté prohledává existující index kořenů slov. Mezi výsledky vyhledávání se pak mohou objevit i dokumenty obsahující přípustné varianty hledaného řetězce. Greenstone standardně nabízí podporu stemmingu pouze pro angličtinu a základně pro francouzštinu, jedním z úkolů této práce bylo zjistit možnost přidání češtiny.

Digitální knihovna Greenstone také nabízí dva odlišné přístupy k budování indexů samotných, které vytváří odlišné struktury, z nichž každá je vhodná pro jiný typ dotazů. Standardním přístupem je využití mechanismu zvaného *MG* (podle knihy *Managing Gigabytes*, viz [50]), která se zabývá technikami na kompresi textů a vytváření struktur indexů pro

efektivní fulltextové vyhledávání). Druhým přístupem je *MGPP*, které je rozšířením *MG* a nabízí navíc například hledání v rámci kontextu (*proximity searching*) nebo prohledávání polí.

Mezi strukturami vytvářenými při budování sbírky hraje důležitou roli již zmiňovaná informační databáze. V ní jsou uchovávány všechny důležité informace týkající se sbírky – její jméno, ikony, unikátní identifikátory dokumentů, asociované soubory (respektive odkazy na ně) a struktury které definují klasifikátory. Všechna data jsou uložena s použitím nástroje *GDBM* (*GNU DataBase Manager*, viz [17]), který implementuje mechanismus uložení párů data/klíč do souborů a poskytuje operace pro manipulaci s nimi. Informační databáze má zásadní význam při procházení obsahu sbírky a slouží jako zdroj informací o sbírce při komunikaci mezi knihovnou a uživatelem.

2.2.3.3 Nástroje pro vytváření sbírek

Celý proces vytvoření sbírky od shromáždění materiálů až po možnost jejího prohlížení sestává z velké řady dílčích kroků. Naprostá většina z nich je díky skriptům napsaným v jazyce Perl zcela automatická a nevyžaduje zásah tvůrce sbírky. Uvedené rozdělení procesu vytváření sbírek na import a budování také odpovídá existenci dvou základních skriptů řídících všechny kroky. Základní postup pro vytvoření nové sbírky je následující:

- Vytvoření adresářových struktur pro umístění souborů sbírky a vytvoření konfiguračního souboru.
- Shromáždění materiálů, které mají být zařazeny do sbírky.
- Přiřazení metadat.
- Úprava konfiguračního souboru sbírky.
- Import materiálů.
- Vybudování sbírky z importovaných materiálů.

Vždy dostupným způsobem je manuální provádění jednotlivých akcí – spuštění skriptů na vytváření nové sbírky, import a budování, kopírování zdrojových dokumentů a úprava konfiguračního souboru pomocí textového editoru. V mnoha případech, zvláště pokud pouze jsou upravovány již existující sbírky nebo vytvářeny podle fungujícího vzoru nové, tento způsob vyhovuje. Z uživatelského pohledu je ale hlavně změna konfiguračního souboru a editace metadat často velice pracná a proto jsou nabízeny dva nástroje, které poskytují přijatelnější alternativu pro vytváření sbírek: *Collector* a *Librarian*. Oba nabízejí grafické uživatelské rozhraní zpřístupňující funkce nabízené různými skripty a sdružují ovládání jednotlivých procesů jednotným způsobem.

Collector

Jednodušším a starším z obou nástrojů je Collector. Nabízí webové rozhraní spolu s nápovědami u jednotlivých kroků vytváření sbírky. Jeho výhodou je jednoduchost a dostupnost i přes síť internet přímo ze stránek knihovny. Díky tomu je možné i na dálku spravovat sbírky nebo vytvářet nové. Rozhraní je ale zároveň největším omezením Collectoru. Je sice možné i na dálku kopírovat soubory a k nim externě přiřazená metadata, ale editace konfiguračního souboru je díky nutnosti editace v textovém poli stejně pracná jako ta manuální. Nástroj Collector je podrobněji popsán v manuálech k systému Greenstone a také v návodu pro uživatele, který vznikl jako součást řešení této práce.

Librarian

Tento nástroj poskytuje více než plnohodnotnou náhradu původního manuálního postupu, neboť řadu nastavení nabízí v přehlednější podobě než jakou je pouhý textový zápis. Librarian je implementován v jazyce Java z důvodů snadné přenositelnosti mezi různými systémy. Na rozdíl od Collectoru je Librarian kvalitním grafickým uživatelským rozhraním určeným ke kompletní správě sbírek, které umožňuje i editaci externě přiřazených metadat podle vybraných metadatových schémat. Většinu nastavení konfiguračního souboru převádí do podoby přepínačů a seznamů s výběrem a stará se o jejich správné uložení. V rámci jednoho okna je tedy možné provést všechny kroky od založení sbírky přes přiřazení metadat a úpravu konfiguračního souboru sbírky až po nastavení parametrů procesů importu a budování a jejich spouštění. Průběh a výsledky tvorby sbírek jsou ukládány do souborů a umožňují tak dohledávat případné chyby.

Librarian je součástí instalace digitální knihovny Greenstone teprve od verze 2.40 a až do dubna roku 2004 k němu neexistoval kromě stručných nápověd oficiální manuál. Podrobný popis nástroje se nachází v přílohách této práce (viz Příloha II) a také ve vypracovaném návodu pro uživatele, který je umístěn na přiloženém CD.

2.2.4 Rozhraní

Jedním z požadavků kladených na digitální knihovny je poskytování takového přístupu k obsahu sbírek a materiálů, které uchovávají, aby bylo umožněno jejich využití v co největší míře. Kvalita organizace a vnitřní struktura vytvořená při budování sbírky je bezcenná, pokud nejsou k dispozici prostředky pro jejich efektivní využití. Pro uživatele by měla digitální knihovna tvořit transparentní celek skrývající veškeré implementační detaily a nabízející informace v snadno uchopitelné podobě.

Obrázek 3: Rozhraní digitální knihovny Greenstone

Greenstone na tyto požadavky reaguje oddělením vrstvy spravující obsah sbírky od vrstvy, která se stará o jeho prezentaci. Při popisu obecné struktury systému (viz kapitola 2.2.2) byly uvedeny dva procesy klíčové pro fungování digitální knihovny Greenstone. Server sbírky při plnění své funkce využívá informací uložených v informačních databázích sbírek, zmíněných v podkapitole o budování sbírky (viz kapitola 2.2.3.2). Popis detailního způsobu zpracování těchto informací přesahuje zaměření této práce, některé jeho podrobnosti jsou zmíněny v literatuře (viz [3]). O zpřístupňování informací získaných dotazováním serveru sbírek se stará

druhá z komponent nazvaná recepční (*receptionist*). Díky oddělení obou komponent je možné měnit způsob prezentace obsahu digitální knihovny i poté, co byly jednotlivé sbírky vytvořeny. Struktury vytvořené ve fázi budování sbírky slouží pro efektivní spravování informací a až na omezení vyplývající z rozhodnutí při návrhu sbírky (jaké druhy indexů budou vytvořeny, jakým způsobem budou převedeny jednotlivé formáty dokumentů) nezasahují do zobrazení uživateli.

Při standardním scénáři, kdy rozhraní mezi digitální knihovnou Greenstone a uživatelem tvoří internetové stránky (viz obrázek 3), využívá recepční pro definování podoby výstupu jednoduchého jazyka textových maker. Ta jsou expandována na základě konkrétního nastavení prostředí až při samotném zobrazení stránek. Obecně platí, že žádná ze stránek digitální knihovny Greenstone není vytvořena předem a uložena na disku, ale je vytvářena dynamicky až na základě požadavku na její zobrazení. Využitím tohoto přístupu lze snadno přizpůsobovat vzhled stránek jednotlivých sbírek stejně jako vytvářet překlady uživatelského rozhraní. Stručný popis jazyka pro definici stránek spolu s příklady je uveden v praktické části této práce (viz kapitola 3.6).

Pro procházení obsahu sbírek a hledání jednotlivých dokumentů jsou použity dva přístupy: *hledání* a *procházení pomocí klasifikátorů*. Hledání splňuje základní požadavek na jakoukoliv spravovanou sbírku informací – možnost získání dokumentů, které obsahují požadovanou informaci. Možnosti nabízené při vyhledávání souvisí s metodou použitou při tvorbě indexů (viz MG a MGPP v kapitole 2.2.3.2) a také s nastaveními zvolenými uživatelem. Standardní prohledávání textu dokumentů a metadat k nim přiřazených je obohaceno o logické operátory a třídění výsledků podle relevantnosti k hledanému řetězci. Rozšířené vyhledávání (s využitím indexů vytvořených pomocí MGPP) pak dovoluje také hledání v kontextu – zadaná slova se mohou nacházet v určitém omezeném rozmezí – a prohledávání podle polí – kombinace omezení na různé atributy dokumentu. U všech typů hledání je také možné uchovávat historii zadaných dotazů a získaných výsledků a také se k nim vracet.

Pokud uživatel nehledá konkrétní informaci, ale má zájem prozkoumat obsah sbírky, případně najít dokumenty související s určitým tématem a nebo navzájem mezi sebou, stává se hledání neefektivním. Pro takové případy nabízí Greenstone zvláštní struktury nazývané *klasifikátory*. Jedná se o zvláštní druhy indexů sloužících pro zobrazení skupin dokumentů seskupených podle zadaných kritérií. Klasifikátory jsou vytvářeny při budování sbírky na základě souborů určujících struktury klasifikátorů a nastavení uložených v konfiguračním soboru.

Obrázek 4: Ukázka hierarchického klasifikátoru

Greenstone nabízí řadu různých klasifikátorů – od jednoduchých seznamů seskupovaných podle abecedy, přes složitější struktury pro organizování podle data, až po hierarchické klasifikátory s téměř neomezeným počtem úrovní. Zcela v duchu přístupu systému Greenstone je možné měnit podobu jednotlivých klasifikátorů (použité ikony, zobrazované údaje), ale i vytvářet klasifikátory vlastní.

Na obrázku 4 je uveden příklad hierarchického dvouúrovňového klasifikátoru, který seskupuje dokumenty podle definovaných kritérií v závislosti na přiřazených metadatech. Na nejvyšší úrovni se nacházejí názvy skupin, na nižší pak názvy jednotlivých dokumentů. Ikona zobrazená vedle názvu slouží pro identifikaci skupiny nebo dokumentu a zároveň pro otevření seznamu obsažených dokumentů nebo jejich obsahu. Na uvedeném obrázku je zobrazena otevřená skupina („Archiv fotografií ÚVT“) a v ní seznam galerií („Celouniverzitní počítačová studovna“, ...). Pomocí klasifikátorů uživatel rychle získá představu o obsahu sbírky z různých pohledů a také je schopen se rychle orientovat i ve velkém počtu dokumentů.

Zobrazení obsahu jednotlivých dokumentů je z části určeno při importu materiálů do sbírky, ale je možné jej pomocí jazyka pro definici stránek (maker) dále přizpůsobovat potřebám konkrétních sbírek. Obecně používané rozhraní internetových stránek umožňuje zobrazení textu a obrázků, pokud používaný prohlížeč obsahuje potřebné zásuvné moduly pak i souborů dalších formátů. Systém Greenstone provádí ve fázi importu dokumentů v opodstatněných případech konverzi jejich obsahu do formy zobrazitelné v prohlížeči. Ačkoliv existují pluginy, které umožňují zpracování i řady proprietárních formátů (například MSWord), často je při konverzi ztraceno důležité formátování. Z těchto důvodů, pro zachování původního dojmu při prohlížení dokumentu, je možné v rámci sbírky uchovávat i původní soubory. Při procházení sbírky je pak uživateli nabízena možnost buď zobrazit původní dokument, nebo jeho do textové podoby převedenou verzi. Tato možnost je využívána také v případech, kdy nelze obsah interpretovat jako text – například u video souborů.

2.2.5 Shrnutí charakteristik systému Greenstone

Informace uvedené v předchozích kapitolách stručně popisují nejdůležitější součásti systému Greenstone a procesy nezbytné pro tvorbu sbírek. Jejich detailní vysvětlení stejně jako pojednání o dalších aspektech tvorby a správy digitálních knihoven Greenstone lze nalézt v literatuře uvedené na konci této práce. V této chvíli, s potřebnými znalostmi základní architektury systému a digitálních knihoven, je možné posoudit, jakým způsobem Greenstone odpovídá požadavkům kladeným na digitální knihovny formulovaným v úvodních kapitolách (viz kapitola 2.1).

Dlouhodobé uchovávání informací je řešeno jejich začleňováním do datových struktur uzpůsobených pro rychlé vyhledávání a přístup. Z hlediska zachování informací v budoucnu interpretovatelné formě je pozitivem využití otevřených (neproprietárních) mechanismů tvorby těchto struktur, které zvyšuje šance na přenositelnost a použitelnost.

Přístup k informacím popisovala kapitola „Rozhraní“ (viz 2.2.4). Využití maker, které bude dále popsáno v praktické části, je dostatečně přizpůsobivým mechanismem pro definování způsobu zobrazení obsahu digitální knihovny. I bez zásahů do předdefinovaného vzhledů poskytuje digitální knihovna Greenstone uspokojivou prezentaci spravovaných informací.

Jednoznačná identifikace dokumentů a mechanismy jejího dosažení, byly popsány v kapitole „Importování materiálů“ (viz 2.2.3.1). Ačkoliv použitá metoda generování identifikátorů v závislosti na obsahu dokumentů má své nevýhody, umožňuje identifikaci i v globálním měřítku. Greenstone dovoluje využívat buď výchozího způsobu generování identifikátorů, který je založen na interním algoritmu, nebo umožňuje tvůrcům sbírek přiřazovat jednotlivým dokumentům identifikátory vlastní. Díky tomu dovoluje flexibilně reagovat jak na požadavky kladené na jednoznačnou identifikaci, tak i vyhovět potřebám uživatelů a umožnit jim vybrat vhodné vlastní identifikační schéma.

Metadata jsou do značné míry využívána v rámci celé digitální knihovny. Dokumenty, které se mají stát základem sbírky, mohou být popsány v rámci externích souborů. Systém pluginů pak umožňuje extrahovat metadata i z jejich obsahu. Na základě získaných metadat jsou vytvářeny indexy a klasifikátory, sloužící pro rychlé vyhledávání a orientaci ve sbírkách.

Interoperabilita s ostatními systémy je řešena změnou chování „recepčního", obecně popsaného v kapitole „Obecná struktura systému Greenstone" (viz 2.2.2). Namísto komponenty komunikující s uživatelem pomocí rozhraní internetových stránek je dosazena komponenta zpřístupňující využití robustního protokolu Z39.50. Díky tomu je umožněna výměna informací i formulace a zodpovídání komplikovaných dotazů s téměř libovolným systémem, který používá stejný protokol. Pro podporu OAI je možné upravit systém tak, aby se stal poskytovatelem dat a umožňoval sběračům metadat získat informace o obsahu knihovny. Protože systém Greenstone je starší než iniciativa OAI, nenabízí zatím možnost vystavování metadat sběračům využívajícím protokol OAI-PMH (*Open Archives Initiative – Protocol for Metadata Harvesting*, viz [35]). Na úpravách, které by zpřístupnily tuto stále rozšířenější formu výměny informací o uchovávaných materiálech, se v současné době za podpory organizace UNESCO pracuje. Digitální knihovnu Greenstone lze také snadno použít jako poskytovatele služeb pro OAI – stačí sběrem získat metadata a pomocí existujícího pluginu (*OAIPlug*) na jejich základě vybudovat sbírku.

Právní, politický a sociální aspekt tvorby sbírek a správy digitální knihovny závisí do značné míry na správci systému. Greenstone nabízí možnost omezení přístupu heslem k určitým sbírkám i k jednotlivým dokumentům. Snadno také lze sbírku, která je předmětem sporu, stáhnout z nabídky digitální knihovny.

2.2.6 Odkazy na zdroje

Informace o systému pro tvorbu digitálních knihoven Greenstone lze nalézt na oficiálních internetových stránkách projektu (viz [20]). Z těchto stránek je možné stáhnout instalace Greenstone pro různé operační systémy (podporovány jsou Windows od verze 3.11 a většina Unixových a Linuxových platforem), manuály pro uživatele i vývojáře systému a odkazy na další stránky.

Hlavní demonstrační stránkou obsahující ukázky mnoha sbírek spravovaných digitální knihovnou je stránka projektu NZDL (viz [31]). Kromě sbírek vytvořených v rámci spolupráce s OSN nabízí i ukázkové sbírky sloužící pro předvedení možnosti knihovny.

Důležitým informačním zdrojem pro uživatele systému Greenstone jsou také elektronické konference, do nichž je možné po zaregistrování přispívat. Díky velkému počtu uživatelů z celého světa a zapojení vývojářů systémů do diskuzí na téma fungování systému Greenstone a problémů při jeho používání, je možné v archivech těchto konferencí nalézt odpovědi na většinu otázek týkajících se tohoto systému pro tvorbu digitálních knihoven. Archivy konference jsou dostupné v podobě sbírek digitální knihovny Greenstone (viz [19]).

2.3 Další volně dostupné systémy

Na stále rostoucí potřebu správy rozsáhlých sbírek digitálních materiálů reaguje řada projektů z celého světa. Jejich cílem je vytvořit kvalitní a snadno využitelná úložiště (*repository*) dokumentů nejrůznějších formátů a pomocí osvědčených služeb pro sdílení metadat, zvláště pomocí protokolu OAI-PMH, umožnit jejich zpřístupnění co nejširšímu okruhu uživatelů. Zatímco digitální knihovna Greenstone nabízí jak správu materiálů, tak i jejich prezentaci, některé z dále popsaných systémů se soustřeďují hlavně na vytváření efektivně

využitelných úložišť a poskytování služeb (vyhledávání, zobrazování) ponechávají na jiných systémech. Informace uvedené v této kapitole a jejích podkapitolách jsou převzaty z přehledu volně dostupných systémů pro správu rozsáhlých úložišť (viz [36]) vytvořeného za podpory nadace SOROS.

Návrh každého z popisovaných systémů odráží původní záměry a požadavky institucí, které na jejich vývoji pracují. Nejedná se tedy o srovnání různých systémů nabízejících stejné služby jako spíše o přehled implementací různých přístupů k řešení problému správy digitálních materiálů. **ARNO** nabízí systém pro centralizovanou správu metadat. **CDSWare** řeší správu velkých úložišť obsahujících různorodé materiály. **DSpace** podporuje shromažďování a prezentaci obsahu v rámci určitých komunit (univerzity, výzkumné ústavy) a nabízí nástroje pro dlouhodobé uchovávání digitálních objektů. **Fedora** poskytuje plnohodnotný systém pro digitální knihovny, který dovoluje obsáhnout i rozsáhlá úložiště digitálních materiálů. Systém **i-Tor** nabízí sadu nástrojů pro vytváření prostředí, ve kterém může být jednotným způsobem zpřístupňován a zobrazován obsah mnoha databází. **MyCoRe** klade důraz na flexibilitu a konfigurovatelnost, která dovoluje využívat různých knihoven a úložišť. V následujících podkapitolách jsou jednotlivé systémy stručně představeny.

2.3.1 ARNO

V rámci projektu ARNO (*Academic Research in the Netherlands Online*, viz [2]) je vyvíjen software pro podporu implementace institucionálních úložišť a jejich provázání s úložišti umístěnými po celém světě (stejně jako s Nizozemskou národní informační infrastrukturou). Projekt je financován organizací IWI (holandská zkratka pro Inovace v poskytování vědeckých informací) a podílí se na něm univerzity v Amsterodamu, Tilburgu a Twente. Výsledný systém je využíván univerzitami v Tilburgu, Amsterodamu, Rotterdamu, Twente a Maastrichtu. Veřejnosti byl systém ARNO dán k dispozici v prosinci roku 2003.

Cíle obsažené v návrhu systému ARNO se liší od zaměření ostatních systémů popsaných v těchto kapitolách. Systém je navržen jako flexibilní nástroj pro vytváření, správu a vystavování archivů a úložišť podporujících OAI. Nabízí centralizovanou tvorbu a administraci obsahů úložišť stejně jako začleňování materiálů poskytovaných koncovými uživateli.

Metadata a příslušné digitální objekty jsou organizovány do archivů (*archives*). Archivy mohou být kombinovány do úložišť (*repositories*), které slouží jako zdroj informací pro sklízení metadat pomocí protokolu OAI-PMH. Modul pro podporu OAI-PMH není limitován použitím metadatového schématu Dublin Core. Pomocí šablon je možné transformovat interní ARNO XML struktury do libovolného formátu. Další vlastnosti systému ARNO zahrnují také schopnost uchovávání různých verzí souborů nebo správu různých vydání (například pracovní materiály před vytisknutím a po něm včetně jednotlivých edicí).

Ačkoliv ARNO jako nástroj pro zpracování obsahu materiálů nabízí značnou flexibilitu, neposkytuje samostatně použitelný systém pro kompletní správu úložišť. Systém je vyvíjen spíše jako sada nezávislých nástrojů pro správu archivů podporujících protokol OAI-PMH a proto nenabízí například rozvinuté uživatelské rozhraní s možností rozsáhlého vyhledávání. Pro zpřístupnění takových služeb je nutné nasadit jiné systémy (například iPort nebo i-Tor).

2.3.2 CDSWare

CDSWare (zkratka pro *CERN Document Server Software*, viz [9]) byl vyvinut na podporu správy dokumentů organizace CERN. Software je udržován a poskytován veřejnosti organizací CERN a podporuje elektronické preprintové archivy, knihovní katalogy vystavované online a

další systémy pro uchovávání internetových dokumentů. CERN využívá CDSWare pro správu více než 450 sbírek obsahujících přes 620 000 bibliografických záznamů a 250 000 fulltextových dokumentů složených z preprintů, časopisů, článků, knížek a fotografií.

CDSWare byl vytvořen s cílem organizování velmi rozsáhlých datových úložišť obsahujících nejrůznější druhy materiálů včetně multimediálních katalogů, popisů objektů muzeí a důvěrných i veřejně dostupných sbírek dokumentů. Každá verze systému je před poskytnutím veřejnosti důkladně testována v prostředí organizace CERN.

2.3.3 DSpace

Systém DSpace (viz [29]) byl vytvořen zejména jako digitální úložiště umožňující uchovávat poznatky získané při výzkumu v mnoha různých oborech. Základní návrh a vývoj systému DSpace probíhal ve spolupráci mezi MIT (*Massachusetts Institute of Technology*) a společností Hewlett-Packard v období mezi březnem roku 2000 a listopadem roku 2002. První verze systému (1.1.1) byla dána k dispozici v srpnu roku 2003.

Struktura DSpace bere ohled na konkrétní zaměření uživatelské komunity. Návrh systému podporuje účast škol, úseků, výzkumných středisek a dalších součástí typických velkých výzkumných institucí. Protože požadavky jednotlivých skupin uživatelů se mohou lišit, dovoluje DSpace využít různých schémat organizace práce a s tím spojených postupů pro zpřístupňování obsahů dokumentů, zajišťování autorizace a zachování intelektuálního vlastnictví. Právě možnosti distribuované správy spolu s nástroji pro podporu plánování uchovávání dokumentů činí systém DSpace vhodný správu úložišť velkých institucí.

DSpace se také zaměřuje na problém dlouhodobého uchovávání výzkumných materiálů. Na výzkumu a vývoji v této oblasti se aktivně podílejí různí uživatelé systému. V budoucnu by tak DSpace mohl sloužit nejen pro ukládání dat a jejich zpřístupňování, ale také pro správu materiálů za účelem jejich archivace.

2.3.4 Fedora

Systém pro správu úložišť digitálních objektů Fedora (viz [39]) je založen na architektuře FEDORA (*Flexible Extensible Digital Object and Repository Architecture*). Je navržen jako základ, na kterém mohou být budovány plnohodnotná institucionální datová úložiště a další digitální knihovny založené na webových technologiích.

Fedora je vyvíjena ve spolupráci mezi University of Virginia a Cornell University. Jedná se o implementaci architektury FEDORA spolu s nástroji, které se starají o správu datových úložišť. Současná verze systému nabízí úložiště schopná efektivně organizovat jeden milión digitálních objektů. Budoucí verze systému poskytnou funkcionalitu potřebnou pro implementace úložišť na úrovni institucí jako je například prosazování specifických pracovních postupů používaných v rámci organizace, správa verzí objektů nebo zlepšení výkonu za účelem správy většího počtu digitálních objektů.

Rozhraní systému je složeno ze tří služeb, založených na webových službách (*web services*):

- *Management API* – definuje rozhraní pro správu úložiště spolu s operacemi nezbytnými pro vytváření a správu objektů klienty (*clients*) .
- *Access API* – stará se o zpřístupňování a poskytování uložených objektů.
- *Access-Lite API* – odlehčená verze Access API, která je implementována jako webová služba dostupná přes protokol http.

Fedora podporuje úložiště různých složitostí – od jednoduchých implementaci, které používají pouze nabízené služby až po vysoce přizpůsobená plnohodnotná distribuovaná digitální úložiště. Nejdůležitějšími charakteristikami projektu Fedora jsou jeho otevřenost, zpřístupnění pomocí webových služeb a využití obecného modelu FEDORA, který dovoluje realizaci různých scénářů práce s digitálními objekty. Fedora ve velké míře využívá XML jak pro práci s metadaty, tak i pro uložení samotných digitálních objektů. Díky tomu je možné snadno exportovat objekty do různých formátů stejně jako poskytnout informace o nich externím systémům. Kromě možnosti přistupovat k obsahu úložišť pomoci webových služeb nabízí také nástroje implementované pomocí jazyka Java, které jsou určeny pro snadnou správu digitálních objektů.

2.3.5 i-Tor

Systém i-Tor (*Tools and technologies for Open Repositories*, viz [22]) byl vyvinut sekcí *Innovative Technology-Applied* (IT-*A*) nizozemského *Institute for Scientific Information Services*. Tvůrci systému jej označují jako „webovou technologii, díky které mohou být různé druhy informací prezentovány pomocí webového rozhraní“ bez ohledu na to, kde a v jakém formátu jsou tyto informace uloženy. Systém i-Tor se zaměřuje na implementaci úložiště „nezávislého na datech“, ve kterém obsah úložiště a funkce uživatelské rozhraní tvoří dvě nezávislé části. V podstatě slouží i-Tor současně jako poskytovatel služeb pro OAI, schopný sbírat metadata z kompatibilních úložišť a databází, ale také jako poskytovatel metadat pro OAI. Systém nabízí organizacím díky možnosti publikování dat z různorodých relačních databází, souborových systémů a internetových stránek velkou svobodu ve způsobu, jakým spravují a organizují své materiály. Dovoluje jak využívat existující relační databáze, tak i vytvářet nové a dále s nimi pracovat.

Díky tomuto přístupu neprosazuje i-Tor použití konkrétních pracovních postupů (*workflow*), ale nabízí organizacím nástroje, které dovolují požadované pracovní postupy v systému zavést (například zabezpečení na různých úrovních nebo oznámení o změnách). Principy využívané systémem i-Tor jej činí vhodným pro použití v organizacích, které potřebují vytvořit zastřešující úložiště založené na již existujících různorodých datových úložištích.

2.3.6 MyCoRe

MyCoRe (viz [30]) vznikl z projektu MILESS univerzity v Essenu. V současné době je vyvíjen konsorciem univerzit za účelem poskytnutí jádra softwarových nástrojů pro podporu digitálních knihoven a archivace (označovaných jako *Content Repositories*, odtud „CoRe“). Tento soubor nástrojů je navržen tak, aby mohl být konfigurován a přizpůsoben lokálním potřebám (proto „My“) bez nutnosti úpravy samotného jádra. Na rozdíl od projektu MILESS, který nabízel pouze datový model Dublin Core, poskytuje MyCoRe datový model, který je kompletně konfigurovatelný. Jádro obsahuje veškerou funkcionalitu potřebnou pro implementaci datového úložiště. Zahrnuje například distribuované prohledávání úložišť rozptýlených po celém světě, podporu OAI, audio/video streaming, správu souborů nebo online správu metadat. MyCoRe nabízí jednoduchou aplikaci využívající funkcionality jádra, která slouží jako příklad, jakým způsobem je možné využívat nabízené jádro pro vytváření vlastních aplikací s využitím metadatových konfiguračních souborů. Systém MyCoRe není závislý na využívání určité databáze. Místo toho poskytuje rozhraní přes vrstvu nazvanou *persistence layer* spolu s implementacemi přístupů k různým databázím.

3 Praktická část

V následujících kapitolách jsou shrnuty praktické výsledky, kterých bylo dosaženo při řešení této diplomové práce. V první podkapitole s názvem „Aplikace pro správu sbírek“ je popsán nástroj, který byl vytvořen pro usnadnění vytváření a správy sbírek systému Greenstone. Kapitoly „Sbírka fotografií z DKF“ a „Sbírka dokumentů různých formátů – Paradox“ se zabývají ukázkovými sbírkami, které demonstrují možnosti nabízené digitální knihovnou Greenstone (některé přístupy použité při prezentaci sbírky fotografií byly v Greenstone použity vůbec poprvé). Kromě popisu sbírek samotných jsou v těchto kapitolách uvedeny také použité postupy pro dosažení prezentovaných výsledků a na příkladech vysvětleny související principy a mechanismy fungování knihovny. V kapitole „Zpřístupnění systému Greenstone českým uživatelům“ jsou popsány výsledky, kterými tato práce přispěla k možnosti použití digitální knihovny Greenstone v českém prostředí a poznatky, které dovolují knihovnu tomuto prostředí dále přizpůsobovat. Kapitola „Pluginy a jejich tvorba“ nabízí ucelený popis významu a způsobů použití modulů určených pro zpracování materiálů sbírek. Zabývá se vysvětlením jejich funkce a popisem jejich struktury spolu s ukázkami nejdůležitějších částí kódu. Poslední kapitola s názvem „Makra“ popisuje další klíčový mechanismus, používaný digitální knihovnou pro prezentaci obsahu sbírek.

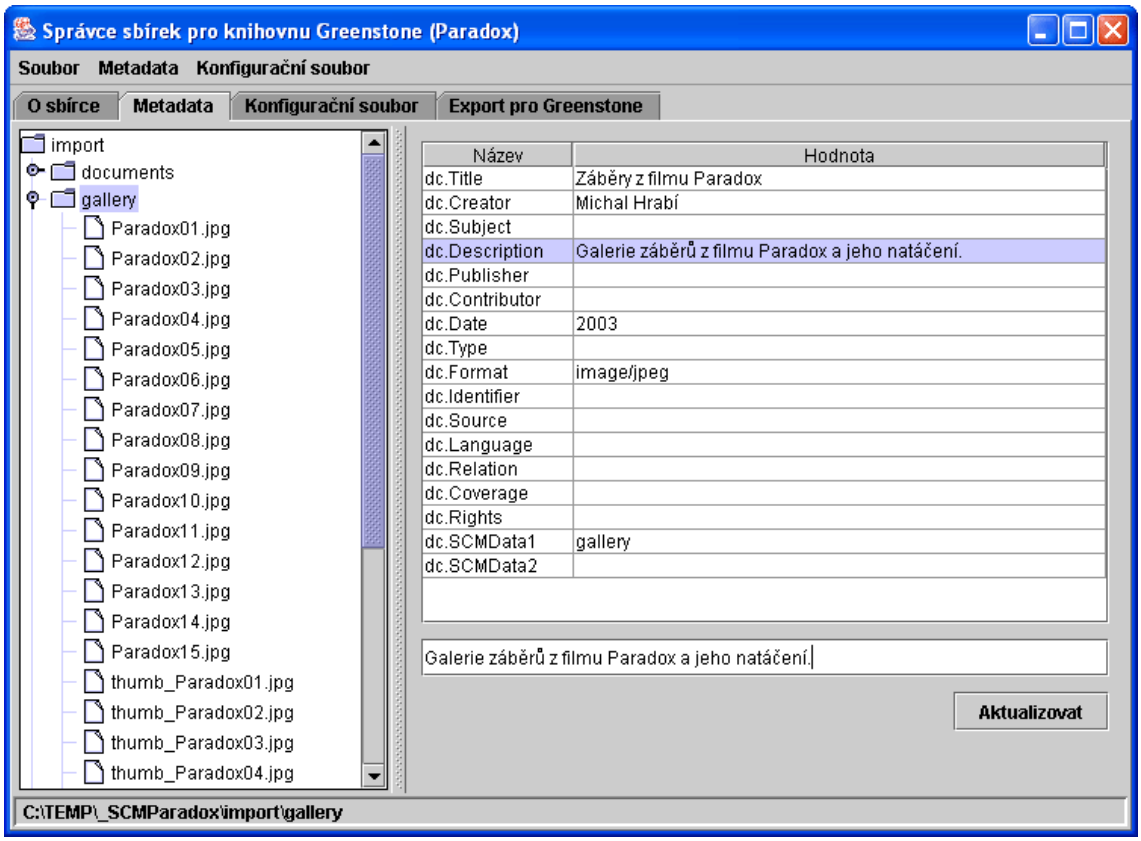
Všechny uvedené výsledky a popisy mechanismů digitální knihovny Greenstone se vztahují k její verzi 2.41, která byla používána při řešení této práce. Ačkoliv se digitální knihovna neustále vyvíjí, vždy je kladen důraz na zpětnou kompatibilitu. V době dokončování této práce byla v souvislosti s realizací dalšího velkého projektu ve spolupráci s organizací UNESCO uvolněna nová verze systému (2.50). Lze však předpokládat, že platnost popisu digitální knihovny a výsledků uvedených v této práci zůstává zachována.

3.1 Aplikace pro správu sbírek

Systém Greenstone dlouhou dobu nenabízel jednotné uživatelské rozhraní, které by umožňovalo přístup k důležitým nastavením využívaným při vytváření sbírek. Uživatel byl nucen pomocí externích nástrojů editovat konfigurační soubory, vytvářet podklady pro hierarchické klasifikátory a manipulovat se soubory určenými pro začlenění do sbírky. Z uživatelského hlediska bylo také velmi pracné vytváření a editace metadat přiřazených externě jednotlivým souborům. Tvůrci systému Greenstone několik let pracovali na nástroji s názvem Librarian (*Greenstone Librarian Interface*, zkracováno na GLI), který splňuje všechny požadavky na správu sbírek pomocí relativně jednoduchého grafického uživatelského rozhraní. První verze nástroje Librarian, jehož základní popis je uveden v Příloze II této práce, však byla uživatelům dána k dispozici až současně s vydáním Greenstone verze 2.40 v polovině roku 2003.

Jedním z úkolů této práce bylo vytvoření jednoduchého uživatelského rozhraní pro správu sbírek, které by dovolilo usnadnit některé z operací prováděných během jejich vytváření a editace i pro méně zkušené uživatele. S využitím programovacího jazyka Java (viz [38]) a vývojového prostředí Eclipse (viz [16]) byla vytvořena aplikace s názvem Simple Collection Manager (SCM), která se kromě editace údajů o sbírce samotné a úprav konfiguračního souboru zaměřuje hlavně na snadnou editaci externích metadat a jejich ukládání ve formátu používaném systémem Greenstone. SCM se na rozdíl od nástroje GLI nesnaží zpřístupňovat všechna nastavení a funkce nabízené digitální knihovnou pro manipulaci se sbírkami. Místo toho

poskytuje uživatelům základní a jednoduchý přístup ke všem důležitým součástem každé sbírky a dovoluje jednoduchou manipulaci s metadaty. Všechny údaje jsou ukládány do interního souboru aplikace, který zajišťuje jejich přenositelnost a uchování všech zadaných informací. Data zadaná v aplikaci SCM lze snadno exportovat do souborů určených pro zpracování systémem Greenstone.



Obrázek 5: Aplikace Simple Collection Manager

V následujících podkapitolách budou popsány důležité funkce aplikace SCM. Obecným popisem použití aplikace, editací obecných údajů o sbírce a úpravami konfiguračních souborů se zabývá kapitola „Nastavení údajů o sbírce“. Správě externích metadat a manipulací se soubory sbírek se věnuje kapitola „Manipulace s metadaty“. Vytváření struktur pro systém Greenstone a jejich ukládání popisuje kapitola „Export metadat pro Greenstone“. Poslední kapitoly pak přibližují některé detaily samotného návrhu aplikace a její implementace a hodnotí dosažené výsledky.

3.1.1 Nastavení údajů o sbírce

Simple Collection Manager je určen pro správu sbírek systému Greenstone. Z tohoto důvodu je jeho fungování přizpůsobeno adresářové struktuře, ve které digitální knihovna Greenstone uchovává informace o svých sbírkách. Po spuštění aplikace SCM je možné v hlavním menu zvolit, zda má být vytvořen nový popis sbírky a nebo má být otevřen již existující (menu „Soubor“, položky „Nový popis sbírky“ a „Otevřít popis sbírky“). Ve stejném menu máme také možnost právě editovanou sbírku uložit nebo zavřít aplikaci. Pro vytvoření nového popisu sbírky je třeba zvolit takový adresář, který obsahuje podadresáře s názvy `etc` a `import`. Požadavek vychází z již zmiňované adresářové struktury používané systémem Greenstone –

adresář `etc` slouží pro ukládání konfiguračního souboru sbírky a dalších souborů s nastaveními, adresář `import` obsahuje materiály určené pro začlenění do sbírky. Po úspěšném založení nového popisu sbírky nebo po otevření již existujícího se zpřístupní záložky sloužící pro editaci údajů a zobrazí se výchozí (v případě otevření existujícího popisu sbírky uložená) data.

Záložka „O sbírce“ je určena pro zadávání obecných informací o sbírce – názvu a popisu sbírky, jejím tvůrci, správci a určení, zda je sbírka ve vývoji a zda je veřejně přístupná. Upravené údaje jsou stisknutím tlačítka Aktualizovat uloženy do interních struktur a dále použity například při generování konfiguračního souboru sbírky. Údaje zadávané na první záložce slouží pro základní identifikaci a popis sbírky v rámci digitální knihovny.

S manipulací s obecnými údaji o sbírce souvisí také editace konfiguračního souboru, kterou zpřístupňuje třetí záložka s názvem Konfigurační soubor. Na ní se nachází jednoduchý textový editor, ve kterém je automaticky nebo na přání uživatele zobrazován konfigurační soubor sbírky. Při otevření nebo založení popisu sbírky je prozkoumán adresář `etc` a pokud obsahuje konfigurační soubor, zobrazí jej SCM v editoru na třetí záložce. Uživatel má možnost obsah souboru libovolně měnit a ukládat jej. Kromě toho může vygenerovat podle šablony výchozí konfigurační soubor sbírky, do nějž budou automaticky doplněny údaje zadané na první záložce. Manipulaci s konfiguračním souborem umožňují položky hlavního menu ve skupině Konfigurační soubor.

3.1.2 Manipulace s metadaty

Hlavním cílem aplikace SCM bylo usnadnit editaci externích metadat a jejich přiřazování jednotlivým souborům sbírky. Systém Greenstone přijímá metadata o budoucích dokumentech sbírky ve formě XML souborů, které jsou uloženy v každém podadresáři adresáře `import`. Tyto soubory obsahují všechna metadata ve formě jednoduše strukturovaných seznamů hodnot jednotlivých atributů. Bez použití dalších nástrojů probíhá přiřazování metadat formou manuální editace jednotlivých souborů. Tento způsob je i díky požadavkům kladeným na strukturu zápisu uživatelsky značně nepohodlný.

SCM nabízí možnost, jak metadata editovat přijatelnější formou, přičemž údaje zadané uživatelem jsou při exportu uloženy na patřičná místa adresářové struktury ve formátu požadovaném systémem Greenstone. Na obrázku 5 je ukázán snímek aplikace SCM s otevřenou záložkou pro manipulaci s metadaty. Na levé straně záložky se nachází strom zobrazující souborovou strukturu adresáře import. Adresáře jsou označeny pomocí ikon složek, soubory pomocí ikon dokumentů. Na pravé straně záložky se nachází tabulka, která pro aktuálně označenou položku souborového stromu nabízí přehled hodnot jednotlivých atributů. Změnou položky adresářového stromu se automaticky aktualizuje i obsah tabulky. Pod tabulkou se nachází editační pole sloužící pro zadávání a úpravu hodnot jednotlivých atributů. V editačním poli se zobrazuje a zpřístupňuje hodnota toho atributu, který je vybrán v tabulce. Tlačítko Aktualizovat pak slouží k přiřazení nové hodnoty aktuálně editovanému atributu. Pomocí procházení adresářové struktury zobrazené ve stromu v levé části okna tak lze prohlížet metadata přiřazená jednotlivým položkám a případně je pomocí editačního pole měnit. Souborový strom je vytvářen dynamicky a jednotlivé větve jsou expandovány až pokud se je uživatel rozhodne otevřít. Díky tomu může být i sbírka obsahující velké množství materiálů ve stromu rychle zobrazena, neboť se načítá jen požadovaná úroveň a není třeba čekat na načtení celé adresářové struktury.

Souborový strom slouží kromě zobrazování obsahu adresáře `import` také k manipulaci se soubory samotnými. Přidávání souborů do adresáře `import` a jejich mazání SCM na rozdíl od nástroje Librarian neřeší. Jedná se však o operace, které lze snadno provést s využitím specializovaných aplikací pro správu souborů. Po zkopírování souborů do adresáře import jsou nové soubory při otevření popisu dané sbírky automaticky zobrazeny. Pokud je při přidávání souborů aplikace SCM spuštěna, lze souborový strom aktualizovat přes kontextové menu (levé tlačítko myši nad stromem a výběr položky Aktualizovat). Mazaní souborů lze provést také pomocí externích nástrojů. Pokud SCM uchovává metadata o položkách, které již nejsou součástí sbírky, je uživatel dotázán, zda mají být odstraněny. Z hlediska uchovávaných metadat je však důležitou manipulací se soubory jejich přesouvání v rámci adresáře `import`. Pokud se uživatel rozhodne, že umístění určitého souboru či adresáře je nevyhovující a rozhodne se jej přesunout, je nutné zachovat všechna metadata přiřazená přesouvaným položkám. SCM umožňuje změnu umístění souborů a adresářů jednoduchým přesouváním položek souborového stromu (mechanismus *Drag and Drop*). Uživatel tedy vybere soubor nebo adresář který chce přesunout, stiskne a podrží levé tlačítko myši a přetažením na nové umístění položky přesune. SCM zároveň zajistí, aby všechna přiřazená metadata zůstala zachována.

SCM dovoluje editovat metadata ve schématu Dublin Core (viz [14]), který je používán pro snadný popis elektronických zdrojů. K 15 základním atributům jsou přidány dva další, určené pro volné použití – například pro uchovávání metadat, na jejichž základě budou generovány hierarchické klasifikátory (viz kapitola 3.1.3). Při přiřazování metadat platí, že potomci uzlu (tj. soubory a podadresáře daného adresáře) v souborovém stromu dědí hodnoty atributů definovaných v uzlu samotném, pokud neurčí hodnoty vlastní. Je-li například adresáři přiřazena hodnota atributu `dc.Title` „Galerie“, všechny položky ležící v souborové struktuře pod tímto adresářem mají tuto hodnotu nastavenu také (v případě, že nemají přiřazenu hodnotu vlastní). Této skutečnosti se s výhodou využívá při zmiňovaném generování hierarchických klasifikátorů.

3.1.3 Export metadat pro Greenstone

Všechny údaje o sbírce i metadata přiřazená k jednotlivým souborům určeným pro začlenění do sbírky uchovává Simple Collection Manager v jednom souboru. Tento způsob umožňuje jednak snadnou přenositelnost údajů o sbírce, ale také dovoluje bezproblémové načítání uložených dat a manipulaci s nimi. Pokud uživatel potřebuje zadané údaje poskytnout systému Greenstone pro vytvoření sbírky, může je exportovat pomocí záložky s názvem „Export pro Greenstone“. Kromě možností uložení konfiguračního souboru a vytvoření souborů s metadaty pro popis souborů nabízí SCM také vytváření podkladů pro generování hierarchických klasifikátorů. Ty umožňují zachycovat strukturu dokumentů a členit je do skupin na různých úrovních hierarchie. Příklad hierarchického klasifikátoru je uveden na obrázku 4 v teoretické části této práce.

Digitální knihovna Greenstone vytváří hierarchické klasifikátory na základě metadat přiřazených souborům a řetězců, které definují strukturu klasifikátorů. Definice struktury jsou uloženy v textových souborech, v nichž jsou uvedeny hodnoty atributů, úroveň hierarchie zadaná pomocí čísel a název, pod nímž má být daná úroveň zobrazována. V následujícím příkladu je uvedena část definice struktury hierarchického klasifikátoru, který definuje 3 úrovně a čtyři skupiny dělení obrázků:

```
„gallery“  1.     „Galerie“
„photo“    1.1.   „Fotografie“
„holiday“  1.1.1. „Fotografie z prázdnin“
„picture“  1.2.   „Obrázky“
```

V konfiguračním souboru sbírky se pro každý vytvářený hierarchický klasifikátor specifikuje jednak atribut, na jehož základě má být struktura vytvořena a také soubor, který strukturu určuje. Při vytváření klasifikátoru je pak u každého dokumentu hodnota vybraného atributu hledána v popisu struktury a na základě čísla definujícího hierarchii je dokument zařazen na správnou úroveň do patřičné skupiny.

Na záložce Export pro Greenstone je možné z nabízeného seznamu vybrat libovolné atributy, na základě jejichž hodnot se poté vytvoří soubory obsahující definice struktur hierarchických klasifikátorů. Soubory jsou uloženy do adresáře `etc` a jejich obsah je zároveň vypsán do informačního panelu spolu s pokyny, jak upravit konfigurační soubor sbírky tak, aby Greenstone na základě metadat vytvořil funkční klasifikátor. Díky tomu lze snadno definovat členění dokumentů do skupin, odpovídajících jejich rozdělení do adresářů. SCM při exportování metadat prochází souborovou strukturu a vytváří definice struktur hierarchických klasifikátorů na základě hodnot vybraných atributů, které jsou v podobě metadat přiřazeny na úrovni adresářů.

Kromě vytváření struktur pro hierarchické klasifikátory dovoluje záložka Export pro Greenstone také generování konfiguračního souboru sbírky, případně ukládání aktuálně editovaného obsahu souboru (na záložce Konfigurační soubor). Poslední a nejdůležitější možnosti je exportování samotných metadat přiřazených souborům a adresářům do XML souborů používaných systémem Greenstone. SCM podobně jako generování hierarchických klasifikátorů prochází souborovou strukturu adresáře `import` a všechna metadata příslušející položkám každého adresáře ukládá do XML souborů, které umisťuje do daných adresářů.

Při exportování je možné vybrat libovolnou kombinaci uvedených možností. Stisknutím tlačítka „Provést export pro Greenstone“ se všechny zvolené kroky provedou. Výsledkem exportu tedy bývají jednak soubory definující externí metadata k souborům určeným pro začlenění do sbírky a také konfigurační soubor sbírky spolu se soubory definujícími struktury hierarchických klasifikátorů. Všechny výsledné soubory jsou uloženy v kódování UTF-8 a umožňují tak zachovat české kódování. Po provedení exportu stačí jen spustit skripty Greenstone pro import materiálů a vytvoření sbírky a výsledná sbírka bude ihned k dispozici v lokální digitální knihovně systému Greenstone.

3.1.4 Implementační detaily

Aplikace Simple Collection Manager byla napsána v programovacím jazyce Java (viz [38]) za pomocí volně dostupného nástroje Eclipse verze 2.1 (viz [16]). Programovací jazyk byl zvolen hlavně z důvodů umožnění snadné přenositelnosti aplikace a nezávislosti na platformě. Testován byl v prostředí *Java Runtime Environment 1.4*, pro vytvoření grafického uživatelského rozhraní byly použity komponenty knihovny SWING. Aplikace samotná, zdrojové kódy, vygenerovaná dokumentace i příručka pro uživatele jsou umístěny na CD, které je součástí této práce.

Při vytváření aplikace byla oddělena vrstva správy dat od vrstvy uživatelského rozhraní. Správu všech údajů o sbírce i metadat přiřazených jednotlivým souborům a adresářům má na starosti třída `CollectionMetadataManager`. Umožňuje jednak načítání souborů popisů

sbírek a jejich ukládání, ale také manipulaci s metadaty a jejich exportování do formátu použitelného digitální knihovnou Greenstone. Veškeré komponenty grafického uživatelského rozhraní slouží pouze pro zpřístupnění funkcí poskytovaných touto třídou. Rozdělení zdrojových kódů do balíků a stručný popis těchto balíků je uveden v následující tabulce:

název balíku	popis
`sgm`	Základní balík. Třída `SimpleCollectionManager`, která patří do tohoto balíku, slouží pro inicializaci celé aplikace.
`sgm.datastructures`	Balík obsahujíc datovou strukturu pro popis uzlů souborového stromu.
`sgm.fileUtils`	Obsahuje třídy pro práci se soubory. Ty umožňují čtení a zápis souborů v kódování UTF-8 a také načítání souborové struktury v abecedním uspořádání.
`sgm.gui`	Balík obsahující všechny třídy používané pro zobrazení grafického uživatelského rozhraní.
`sgm.MDStructures`	Obsahuje třídu `CollectionMetadataManager`, která slouží pro správu všech metadat a údajů sbírky. Dále obsahuje třídu `ItemMD`, používanou pro uchovávání metadat k souborům a adresářům.
`sgm.resources`	Externí soubor obsahující všechny textové řetězce používané pro komunikaci s uživatelem.

tabulka 1: Rozdělení zdrojových kódů

Třída `CollectionMetadataManager` uchovává údaje o sbírce samotné a také metadata o jednotlivých souborech určených pro začlenění do sbírky. Tato metadata jsou spravována instancemi třídy `ItemMD`, které současně poskytují některé základní funkce pro jejich zpřístupňování a starají se o správný zápis metadat do souborů. Všechny instance třídy `ItemMD` patřící k souborům editované sbírky jsou uchovávány v asociativní struktuře, která na základě klíčů umožňuje efektivně vyhledávat hodnoty (`HashMap`). Klíčem je umístění souboru v adresářové struktuře, hodnotou pak samotná metadata. Při konstruování souborového stromu jsou pro každou položku na základě jejího umístění v adresáři `import` hledána příslušná metadata v asociativní struktuře. Pokud jsou nalezena, naváží se na konkrétní uzel stromu, čímž v budoucnosti umožní jejich rychlé zpřístupnění a editaci. V kapitole popisující manipulaci s metadaty (viz 3.1.2) bylo uvedeno, že konstruování souborového stromu probíhá dynamicky. Proto i navazování metadat probíhá až ve chvíli, kdy uživatel poprvé zobrazuje uzly stromu. Díky tomu je možné rychle otevírat a editovat i rozsáhlé sbírky – všechny operace provazování uzlů stromu s metadaty se provádějí až ve chvíli, kdy je potřeba údaje zobrazit.

Export metadat a vytváření hierarchických klasifikátorů vyžaduje inicializaci celé stromové struktury. Při obou operacích je postupně rekurzivně procházen adresářový strom, jsou analyzována metadata navázaná na jeho jednotlivé uzly a výsledky jsou ukládány do souborů. Kompletní expanze stromu a navázání všech metadat také umožňuje zjistit, zda nejsou uchovávána metadata o již neexistujících položkách. Pokud počet nevyužívaných metadatových struktur přesáhne určitou mez, je uživatel dotázán, zda mají být zastaralé údaje odstraněny.

Další implementační detaily i vygenerovanou dokumentaci a uživatelskou příručku je možné nalézt na CD přiloženém k této práci.

3.1.5 Zhodnocení aplikace

Cílem aplikace Simple Collection Manager bylo zpřístupnit některá důležitá nastavení a editaci metadat při vytváření a správě sbírek digitální knihovny Greenstone. SCM umožňuje rychlý popis nových sbírek a v rámci jednotného uživatelského rozhraní dovoluje editovat údaje o sbírce samotné, upravovat konfigurační soubor sbírky, přiřazovat metadata a vytvářet soubory, které jsou dále použitelné systémem Greenstone. Důraz byl kladen hlavně na co nejsnadnější manipulaci s metadaty. Díky přehledné stromové struktuře a zobrazení údajů v tabulce je uživatel ušetřen procházení jednotlivými soubory obsahujícími metadata. Exportování zadaných údajů pak umožňuje snadno vytvářet a aktualizovat důležité řídící struktury. Celý proces vytvoření sbírky lze provést v následujících krocích:

- vytvoření struktury nové sbírky (například pomocí skriptu `mkcol.pl`)
- editace údajů a metadat pomocí aplikace SCM
- export metadat pro Greenstone
- spuštění skriptu na import dokumentů (`import.pl`)
- spuštění skriptu na vytvoření sbírky (`buildcol.pl`)

Ačkoliv byl mezitím vytvořen nástroj Librarian, který nabízí více možností nastavení a editace parametrů, může aplikace Simple Collection Manager posloužit pro rychlý popis zdrojových materiálů a vytváření hierarchických klasifikátorů i méně zkušenými uživateli systému Greenstone. Volně dostupné a okomentované zdrojové kódy také mohou pomoci případným zájemcům o vytvoření podobných nástrojů nabízejících další prostředky pro správu sbírek v rámci digitální knihovny Greenstone.

3.2 Sbírka fotografií z DKF

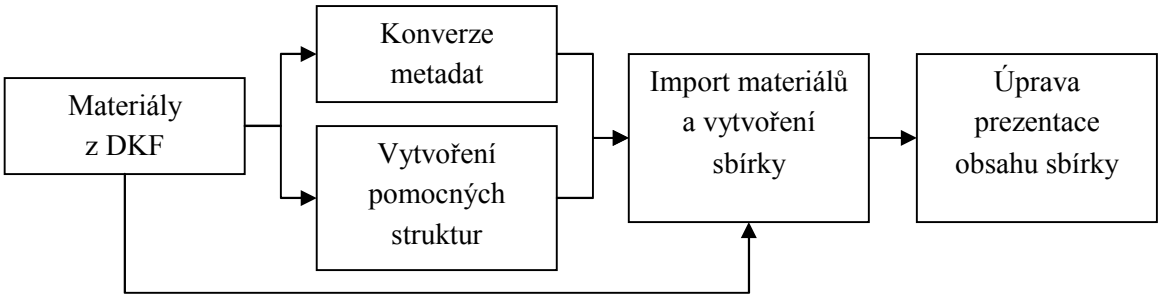
Pro správu a prezentaci digitálních a digitalizovaných fotografií z archivů Masarykovy Univerzity v Brně je v rámci Ústavu výpočetní techniky (ÚVT) vyvíjen systém DKF (Digitální Katalog Fotografií, viz [5]). Umožňuje hierarchicky organizovat velké množství fotografií, přidávat k nim metadata a výsledek nabídnout uživateli ve formě galerie s možností procházení různých úrovní (složky/fotografie) a zobrazování fotografií v různých velikostech (od náhledu až po archivní soubor s velkým rozlišením). Kromě toho nabízí řadu dalších služeb pro správce i uživatele, které například dovolují vzdálenou správu sbírek, omezování přístupu k materiálům pouze pro oprávněné osoby, vytváření seznamů oblíbených položek nebo zasílání fotografií elektronickou poštou.

Ačkoliv je systém DKF stále ve vývoji, umožňuje v současné době vytvářet a spravovat sbírky fotografií a dovoluje exportovat uchovávaná metadata a údaje o struktuře ve formě XML souborů. Přesto však jeho největším omezením zůstává relativně úzké zaměření na správu fotografií a obrazových dokumentů. V případě potřeby organizovaného uchovávání elektronických dokumentů jiných formátů (text, audio, video) by bylo nutné zapojit jiné systémy. Z hlediska dlouhodobé správy materiálů je však vhodnější stavět na jednotném přístupu, protože využívání různorodých specializovaných systémů ztěžuje jednotnou prezentaci obsahu a hlavně činí celek velice obtížně udržovatelný a dále přenositelný.

Systém pro tvorbu digitálních knihoven Greenstone se zaměřuje na správu dokumentů nejrůznějších formátů a nabízí zmiňovaný jednotící přístup. Velká obecnost však často přináší nutnost omezení některých funkcí, které mohou být nabízeny specializovanými systémy. Jedním z úkolů této práce bylo vyzkoušet možnost použití systému Greenstone pro správu a prezentaci sbírek fotografií se zachováním důležité funkčnosti nabízené systémem DKF. Ta zahrnuje především možnost organizace fotografií do tematicky souvisejících skupin, asociování metadat a zpřístupnění materiálů uživatelům ve formě galerií s možností procházení a zobrazování různých náhledů fotografií.

3.2.1 Obecný postup vytvoření sbírky

Základem pro vytvoření sbírky v systému Greenstone byly fotografie v různých velikostech a metadata exportovaná z existující sbírky spravované systémem DKF. Konkrétně se jednalo o soubor fotografií zabývajících se historií ÚVT. Metadata byla uložena ve formátu XML podle schématu používaného v DKF, fotografie ve formátu `jpg`.



Obrázek 6: Schéma postupu tvorby sbírky

Na obrázku 6 je schématicky naznačen celý postup při vytváření sbírky fotografií. Nejprve musela být upravena metadata získaná z DKF. Jednalo se jednak o konverzi metadat do schématu XML používaného pro popis zdrojových materiálů v systému Greenstone a jednak o vytvoření pomocných struktur sloužících pro zobrazení galerií. Tento krok je podrobně rozebrán v kapitole „Vytvoření metadatových struktur“.

Ve druhé fázi bylo nutné na základě upravených metadat a původních fotografií vytvořit sbírku digitální knihovny Greenstone. Požadavky na způsob prezentace byly natolik nestandardní, že bylo nutné vytvořit nový plugin, který zpracovává zdrojové materiály požadovaným způsobem. Začleněním materiálů do sbírky se zabývá kapitola „Import materiálů a vytvoření sbírky“.

Na závěr bylo třeba zasáhnout do způsobu prezentace obsahu. Kromě rozdílného zobrazení dokumentů na různých úrovních hierarchie (galerie/fotografie) bylo také potřeba umožnit snadnou navigaci mezi jednotlivými fotografiemi i celými sbírkami a nabídnout možnost prohlížení různých náhledů. Popisem řešení těchto problémů se zabývá kapitola „Úprava uživatelského rozhraní“.

3.2.2 Vytvoření metadatových struktur

Struktura metadat exportovaných z DKF odráží způsob organizace materiálu ve sbírkách. Základními jednotkami jsou `instance` – ty odkazují přímo na fyzické soubory a uchovávají základní údaje o jejich vlastnostech (výška a šířka). `Fotka` seskupuje více souvisejících instancí (stejná fotografie v různých rozlišeních) a uchovává informace o názvu fotografie, její

popis a další relevantní údaje. Na nejvyšší úrovni organizace se nachází *složka*, která seskupuje tématicky související fotografie (například „Celouniverzitní počítačová studovna“). Složce je, podobně jako fotce, možné přiřadit název a další popisná metadata.

```
<sofsem id="00200009.000" pred="00200014.000" nasl="00200005.000">
    <nazev>Instalace EC-1033</nazev>
    <popis>Instalace prvního velkého počítače univerzity, počítače
        EC-1033, v prostorách Láboratoře počítacích strojů VUT
        Brno na Třídě Obránců míru (dnešní Údolní ulice).</popis>
    <fotka id="00200009.001" nasl="00200009.002">
        <nazev>Instalace EC-1033</nazev>
        <popis>Instalace snímače štítků.</popis>
        <instance id="00200009.001.1" pred="00200009.001.2">
            <soubor>00200009.001.1.jpg</soubor>
            <horiz>1024</horiz>
            <vert>714</vert>
        </instance>
        <instance id="00200009.001.2" pred="00200009.001.3"
                                      nasl="00200009.001.1">
            <soubor>00200009.001.2.jpg</soubor>
            <horiz>640</horiz>
            <vert>446</vert>
        </instance>
        <instance id="00200009.001.3" nasl="00200009.001.2">
            <soubor>00200009.001.3.jpg</soubor>
            <horiz>160</horiz>
            <vert>112</vert>
        </instance>
    </fotka>
...
```

Obrázek 7: Ukázka metadat exportovaných z DKF

Kromě popisných metadat jsou však uchovávána i metadata zachycující strukturu sbírek. Každý prvek hierarchie má přidělen svůj jednoznačný identifikátor (označovaný `id`). Díky tomu je možné uchovávat údaje o řazení materiálů a konstruovat navigaci dovolující přecházet mezi jednotlivými náhledy jedné fotografie, procházet postupně složku po fotografiích a nebo procházet celými složkami. Potřebné informace o řazení materiálů na jednotlivých úrovních jsou uchovávány v atributech `pred` a `nasl` a mají podobu identifikátorů odkazovaných dokumentů. Na obrázku 7 je uveden příklad části souboru, který zachycuje většinu popsaných prvků. Jedná se o popis složky (ta je označena elementem `sofsem`) obsahující fotografie související s instalací počítače EC-1033. Pro ilustraci je uveden i záznam popisující první fotografii a její instance.

Digitální knihovna Greenstone využívá pro popis zdrojových materiálů metadatové schéma, které je z hlediska možnosti vnořování elementů „ploché“. Jelikož má za úkol postihnout metadata, která mohou být přiřazena libovolným souborům a vyžívána pro různé účely, musí poskytovat univerzální řešení. K libovolnému souboru nebo skupině souborů dovoluje přiřadit metadata v podobě výčtu hodnot jednotlivých atributů, umístěných do navzájem nevnořených elementů. Pro dosažení požadované funkčnosti v rámci budoucí sbírky tedy bylo nutné vstupní

metadata (strukturovaná a vnořovaná) vhodně transformovat do jednodušší struktury používané systémem Greenstone. Řešením tohoto problému se zabývá kapitola „Transformace metadat“.

Dalším problémem je přiřazení metadat prvkům, které nemají fyzický protějšek – soubor na disku. Jedná se například o údaje o složce, která slouží pouze jako kontejner sdružující fotografie. Po konzultaci s tvůrci systému Greenstone byl zvolen postup vytvoření pomocných struktur definujících galerie, na které jsou navázána metadata příslušející složkám. Využitím těchto struktur také bylo dosaženo zobrazení galerií, které více než standardní klasifikátory nabízené Greenstone odpovídá původní podobě sbírek DKF. Tvorbou pomocných struktur se zabývá kapitola „Pomocné struktury“.

3.2.2.1 Pomocné struktury

Omezení kladená způsobem asociování metadat s materiály určenými pro začlenění do sbírky digitální knihovny Greenstone přinesla potřebu vytvoření pomocných struktur, které slouží pro navázání metadat na úrovni složek. Zároveň jich je využíváno pro zobrazení náhledů fotografií ve složkách uložených. Pro vytvoření souboru definujícího galerie je použito transformační šablony (lze ji nalézt na přiloženém CD-ROM), která z původního popisu obsahu složky vytvoří jednoduchý strukturovaný XML soubor.

```
<Gallery>
      <GalleryID>00200009.000</GalleryID>
            <Photo>
                  <PhotoID>00200009.001</PhotoID>
                  <PhotoTitle>Instalace EC-1033</PhotoTitle>
            </Photo>
...
</Gallery>
```

Obrázek 8: Ukázka obsahu souboru popisujícího galerii

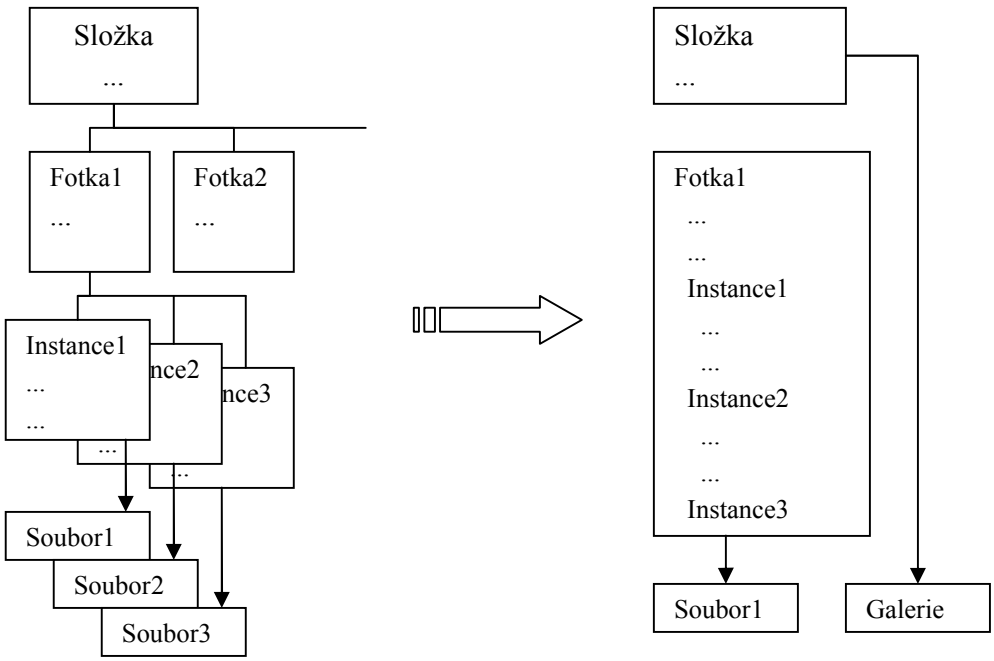
Každá galerie je určena pouze identifikátorem (jedná se o původní identifikátor složky) a výčtem fotografií – každá je určena pouze svým identifikátorem a názvem. Výsledkem použití transformační šablony na soubor s metadaty získaný exportem z DKF je tedy nový XML soubor obsahující popis obsahu jednotlivých galerií. Takto získaný soubor je nakonec pomocí skriptu napsaného v jazyce Perl rozdělen do menších souborů, z nichž každý obsahuje definici právě jedné galerie. Výsledné soubory jsou při importování materiálů dále zpracovány, provázány s fotografiemi a metadaty a slouží jako základ pro vytvoření stránek jednotlivých galerií (více viz kapitola 3.2.3).

3.2.2.2 Transformace metadat

Při vytváření sbírky fotografií hrál důležitou roli fakt, že identifikátory složek, fotek a instancí ve sbírkách DKF jsou vytvářeny hierarchicky (z identifikátoru instance lze určit ke které patří fotce, stejně tak u každé fotky lze určit nadřazenou složku). Názvy fyzických souborů s fotografiemi pak přímo odpovídají identifikátorům instancí, které na ně odkazují. Díky tomu mohla být vytvořena transformační šablona, která umožnila zachovat nejen metadata popisná, ale i strukturální, určená pro navigaci mezi fotografiemi a složkami. Díky změně mechanismu přidělování unikátních identifikátorů, která bude popsána v jedné z následujících

kapitol (viz 3.2.3), jsou původní identifikátory získané z DKF využívány i v rámci knihovny Greenstone.

Jak již bylo uvedeno, Greenstone umožňuje asociovat metadata pouze s existujícími soubory nebo skupinami souborů. Soubory obsahující pomocné struktury (galerie), jejichž vytváření bylo naznačeno v minulé kapitole, dovolují navázat metadata příslušejícím původním složkám v DKF. Podobným způsobem bylo potřeba vyřešit navázání metadat spojených s fotkami. Místo vytváření dalších pomocných struktur se v rámci transformace dávají dohromady metadata pro fotku s metadaty pro jednotlivé instance a celá tato metadatová struktura se asociuje se souborem odkazovaným jednou z instancí (konkrétně je používán soubor s nejvyšším rozlišením). Schematicky je transformace znázorněna na následujícím obrázku:



Obrázek 9: Transformace struktury metadat

Veškerá metadata exportovaná z DKF jsou naznačeným způsobem převedena do formátu akceptovaného systémem Greenstone. V průběhu transformace jsou navíc doplněny další údaje, které při procházení sbírky umožňují kombinovat navigaci převzatou z DKF s klasifikátory a vyhledáváním systému Greenstone. Na úrovní složek i galerií je přidán atribut uchovávající pořadí (`dc.Order`). Dokumenty organizované pomocí klasifikátorů jsou díky pořadí seřazeny způsobem stejným, jakým je lze procházet pomocí šipek „předchozí“ a „následující“. Na úrovni složek je doplňován atribut `dc.Gallery`, který určuje umístění dané složky do patřičné galerie (např. galerie „Fotoarchiv ÚVT“) a slouží pro správné zobrazení hierarchického klasifikátoru galerií. K metadatům popisujícím fotky je navíc přidán atribut `dc.GroupIdentifier`, později používaný pro možnost návratu do galerie nadřazené prohlížené fotografii. Ukázka výsledného transformovaného souboru je umístěna v Příloze I této práce, transformační šablona a další související materiály se nacházejí na přiloženém CD.

3.2.3 Import materiálů a vytvoření sbírky

Význam a průběh fáze importování materiálů byl popsán v teoretické části této práce. Jedná se o zpracování souborů, které se mají stát součástí sbírky. Během importu bývá prováděna

konverze obsahu souborů, asociování metadat a vytváření struktur, které jsou základem pro novou sbírku. Ačkoliv součástí instalace systému Greenstone je řada pluginů umožňujících zpracování nejrůznějších formátů, pracují většinou spíše odděleně a jejich jediným úkolem je převést soubory a metadata do formy dále použitelné v digitální knihovně. V případě sbírky DKF však bylo zapotřebí zvolit komplexnější přístup a místo jednoduché konverze obsahu zajistit fungující propojení jednotlivých dokumentů tak, aby byla možná navigace na různých úrovních a správné zobrazování popisných metadat. Z těchto důvodů byl v rámci této diplomové práce vytvořen nový plugin nazvaný *DKFPlug* (nachází se na CD přiloženém k této práci). DKFPlug zpracovává soubory obsahující definice galerií (opatřené příponou `dkf`) a všechny grafické soubory (přípony `jpg`, `gif`, `bmp` a další).

V předchozích kapitolách byl popsán význam souborů definujících galerie jako pomocných struktur umožňujících zachování metadat přiřazených složkám. Kromě toho ale na základě obsahu těchto souborů vytváří DKFPlug stránky s náhledy fotografií patřícími do dané galerie, které zároveň slouží jako odkazy na fotografie samotné. V rámci sbírek knihovny Greenstone je pro navigaci mezi dokumenty možné použít „relativních odkazů," které jsou obdobou URL adres. Ve formě atributu přiřazeného dokumentu ve fázi importu lze tímto způsobem určit adresu dokumentu v dané sbírce. DKFPlug využívá tohoto principu pro zpřístupnění procházení mezi fotografiemi a galeriemi. Každému dokumentu je přiřazena unikátní adresa složená z umístění zdrojového souboru v adresáři `import` a jeho identifikátoru. Z hodnot atributů `pred` a `nasl` získaných z původních sbírek DKF jsou známy identifikátory předchozího i následujícího dokumentu, při transformaci byl navíc ke každé fotografii přidán atribut `dc.GalleryIdentifier` (viz předchozí kapitola o transformaci metadat) uchovávající identifikátor nadřazené galerie. Na základě těchto údajů se při procházení sbírkou fotografií určují adresy dokumentů a díky nim tak funguje většina navigace.

Při zpracování souboru obsahujícího definici galerie vytváří DKFPlug pomocí HTML formátování dokument, který obsahuje náhledy fotografií dané galerie zarovnané do tabulky. Odkazy umožňující zobrazit libovolnou z nabízených fotografií jsou odvozovány na základě identifikátorů fotografií uložených v souboru definujícím galerii (viz ukázka v kapitole 3.2.2.1). Aby bylo možné zobrazit náhledy fotografií, bylo nutné upravit způsob, jakým Greenstone generuje unikátní identifikátory dokumentů. Soubory asociované s importovanými dokumenty (například obrázky) jsou totiž při importu ukládány do adresářů, jejichž jména odpovídají unikátním identifikátorům dokumentů. Pro získání souboru asociovaného s jiným než aktuálně prohlíženým dokumentem je třeba znát přesný identifikátor daného dokumentu a na jeho základě odvodit umístění souboru. Většinou jsou identifikátory při importu generovány jako otisky (*hash*) souborů a jako takové je obtížné je za běhu knihovny určit. Z tohoto důvodu bylo generování identifikátorů nahrazeno jejich přiřazováním – v rámci zpracování souboru pluginem totiž lze stanovit, jaký způsob bude použit. V případě sbírky fotografií bylo využito již existujících identifikátorů používaných sbírkami DKF. Díky tomu je možné na základě znalosti identifikátoru dokumentu určit přesné umístění asociovaných souborů a tak zobrazit náhledy fotografií ve vytvořených galeriích.

Při zpracování souboru grafického formátů nejprve DKFPlug zjistí, zda jsou k souboru asociována metadata. Pokud ano, jedná se o „reprezentanta", který byl při transformování metadat vybrán jako zástupce fotografie a všech jejích instancí (viz předchozí kapitola o transformaci metadat) a je pluginem dále zpracován. V opačném případě se jedná buď o soubor příslušející jedné z instancí, která nebyla vybrána jako „reprezentant", a nebo soubor nemá být

zařazen do sbírky – v obou případech je ignorován. Při zpracování souboru se nejdříve vytvoří zástupný dokument systému Greenstone, jemuž se přiřadí identifikátor odpovídající identifikátoru z DKF. Dále se k dokumentu přiřadí atribut určující jeho adresu v rámci sbírky (adresa je opět založena na identifikátoru, jak bylo popsáno v této kapitole) a poté se přidají všechna popisná a strukturální metadata získaná z DKF a během transformace. Na závěr jsou dokumentu asociovány všechny soubory odkazované jednotlivými instancemi dané fotografie.

Výsledkem importu zpracovávaného pluginem DKFPlug jsou dva druhy interních dokumentů knihovny Greenstone. Prvním druhem jsou dokumenty zastupující fotografie, které kromě metadat uchovávají i grafické soubory odpovídající původním instancím. Druhým typem dokumentů jsou galerie, které uchovávají metadata o původních složkách, a zároveň obsahují HTML stránky zpřístupňující náhledy všech fotografií dané galerie a odkazy na ně. Na obou úrovních, stejně jako mezi nimi navzájem, funguje navigace pomocí interních odkazů – je možné získat předchozí či následující fotografii nebo galerii a stejně tak je možné přecházet z galerií na náhledy fotografií a naopak. Způsob zobrazování jednotlivých velikostí fotografií (tj. různých instancí téže fotografie v různých rozlišeních), je zmíněn v následující kapitole s názvem „Úprava uživatelského rozhraní“.

Samotné vytvoření sbírky už je ovlivněno jen nastavením konfiguračního souboru. Veškeré potřebné informace jsou popsaným způsobem zpracovány a uchovány během importu, budování sbírky pouze na jejich základě vytvoří indexy a další struktury potřebné pro fungování sbírky v rámci knihovny Greenstone.

3.2.4 Úprava uživatelského rozhraní

Posledním krokem potřebným pro zprovoznění sbírky fotografií z DKF byla úprava způsobu její prezentace. K dosažení požadované funkčnosti byly využity jednak standardní prostředky nabízené systémem Greenstone (vyhledávání a klasifikátory) a také možnost dynamického ovlivňování vzhledu a chování pomocí jazyka maker (viz kapitola 3.6). Při zobrazení sbírky fotografií z DKF má uživatel tři možnosti přístupu ke galeriím a fotografiím:

- Vyhledávání podle názvů nebo popisů.
- Procházení abecedním seznamem dokumentů (galerií a fotografií).
- Procházení hierarchického klasifikátoru galerií (galerie sloučené do skupin podle jednotících charakteristik – například galerie spadající pod Archiv fotografií ÚVT).

Nabízené přístupy umožňují najít konkrétní dokument na základě znalosti jeho názvu nebo popisu a také procházet obsah sbírky ze dvou různých pohledů. Byly provedeny úpravy způsobu zobrazování výsledků vyhledávání i jednotlivých klasifikátorů tak, aby odpovídal sbírce fotografií – každý dokument je zastoupen náhledem (v případě fotografie) nebo zástupnou ikonou (v případě galerie) a názvem. Náhledy zároveň slouží jako odkazy na dokumenty samotné. Zásadní změny bylo třeba provést až ve způsobu zobrazování fotografií a galerií. Bylo nutné zajistit správné zobrazení různých náhledů téže fotografie se zachováním zobrazení příslušných metadat a také vytvořit fungující navigaci mezi dokumenty.

3.2.4.1 Zobrazení různých náhledů

Aby bylo možné na přání uživatele zobrazit fotografii v různých rozlišeních (náhled/pro prohlížení v okně/velká velikost), bylo nutné zasáhnout do způsobu, jakým jsou vytvářeny a

vyhodnocovány odkazy používané digitální knihovnou Greenstone pro zobrazení stránek s dokumenty. Všechny parametry obsahující nastavení prostředí (použitý jazyk, kódování, prováděná akce, stránka ke zobrazení a další) jsou předávány jako součást adresy zadávané prohlížeči sloužícímu pro zobrazení rozhraní digitální knihovny. Parametry jsou předávány ve formě `cgi` argumentů a vytvářeny automaticky systémem na základě výchozích nastavení a aktuálně prováděných akcí. Digitální knihovna parametry při každém požadavku na zobrazení stránky vyhodnotí a na jejich základě vrátí prohlížeči požadovaný dokument. Greenstone nabízí možnost předávat touto formou parametry vlastní, které je možné použít například pro bližší určení způsobu zobrazení dokumentů – tímto způsobem bylo ve sbírce fotografií DKF dosaženo zobrazování různých náhledů téže fotografie bez nutnosti vytvářet pro každý náhled zvláštní dokument.

Zavedení nového parametru se dosáhne jeho definováním v hlavním konfiguračním souboru digitální knihovny (soubor `main.cfg` v podadresáři `etc` instalace knihovny Greenstone):

```
cgiarg          shortname=dmode longname=documentmode \
                multiplechar=false argdefault=2 \
                defaultstatus=weak savedarginfo=must

pageparam         dmode 0
```

Obrázek 10: Deklarace nového parametru v souboru main.cfg

Klíčové slovo `cgiarg` zavádí nový parametr s názvem `documentmode`, určuje jeho zkrácený tvar, výchozí hodnotu, počet znaků a povinnost. Řádek uvozený `pageparam` dovoluje použít nově definovaný argument jako parametr stránky a za běhu zjišťovat jeho hodnotu pomocí maker (bude popsáno v kapitole „Navigační panel“). Bližší informace o významu a použití parametrů lze nalézt v literatuře (viz [3]). Zároveň s definováním nového parametru byl vytvořen i jednoduchý soubor maker umožňující zjišťování aktuální hodnoty tohoto parametru a na jejím základě zobrazení požadovaného náhledu (makra jsou umístěna v souboru `arguments.dm` v adresáři `macros`).

3.2.4.2 Navigační panel

Od transformace metadatových struktur exportovaných z DKF přes importování pomocí speciálního pluginu DKFPlug až po definování vlastních parametrů probíhala postupná příprava podkladů, které kromě jiného slouží pro vytvoření navigace mezi dokumenty. Jednotná navigace v rámci galerií a fotografií sbírky DKF je sdružena do oblasti, která bude dále označována jako navigační panel. Na něm má uživatel k dispozici jednak funkce nabízené systémem Greenstone (vypnutí zvýrazňování hledaného textu a otevření dokumentu v novém okně) a také tlačítka umožňující přejít na předchozí/následující dokument (na úrovni galerií i fotografií), zobrazovat fotografii v různých náhledech nebo přejít do galerie nadřazené právě prohlížené fotografii.

Obrázek 11: Navigační panel

Na obrázku 11 je zobrazen navigační panel se všemi dostupnými tlačítky. Vlevo jsou umístěna tři tlačítka pro zobrazení různých náhledů téže fotografie. Uprostřed panelu se nachází

šipky pro přechod na předchozí/následující dokument a skok na nadřazenou galerii. Vpravo jsou pak umístěny odkazy nahrazující standardní tlačítka pro otevření dokumentu v novém okně a vypnutí zvýrazňování hledaného textu. Podoba navigačního panelu zůstává stále stejná, pouze nedostupná tlačítka jsou v závislosti na prohlíženém dokumentu skrývána (například zobrazování různých velikosti fotografií na úrovni galerií).

Odkazy na předchozí a následující dokumenty jsou vytvářeny dynamicky při zobrazování dokumentů na základě atributů získaných z DKF a přidaných ve fázi importu. Například odkaz na předchozí dokument je v konfiguračním souboru sbírky definován takto:

```
_dkfprevarrow_('_httpextlink_&rl=1&href=[URLPREFIX][dc.Previous]
                &dmode=_getdmode_')
```

Obrázek 12: Ukázka definice odkazu na předchozí dokument

Řetězce uzavřené mezi podtržítka jsou názvy maker, řetězce uzavřené do hranatých závorek odkazují na atributy aktuálního dokumentu (metadata). Definici odkazu na obrázku 12 je možné interpretovat následujícím způsobem: „Vytvoř šipku odkazující na předchozí dokument (`_dkfprevarrow_` je makro nadefinované pro potřeby sbírky DKF a slouží k umístění šipky odkazující na předchozí dokument) . Půjde o standardní odkaz v rámci systému Greenstone (systémové makro `_httpextlink_`) a bude relativní vůči sbírce (argument `rl=1` vyjadřuje, že odkaz nevede mimo sbírku). Adresa odkazovaného dokumentu je složena z těchto částí:

- `URLPREFIX` – atribut generovaný při importu, který určuje, jakou cestu sdílejí všechny dokumenty jedné galerie.
- `dc.Previous` – identifikátor předcházejícího dokumentu.

Při přechodu na odkazovaný dokument zároveň předávej argument `dmode` s hodnotou odpovídající hodnotě aktuální (`dmode` je používán pro určení toho, jaký náhled fotografie má být zobrazen; `_getdmode_` je makro sloužící pro zjištění aktuální hodnoty tohoto parametru).“

Podobným způsobem jsou vytvářeny odkazy na následující dokument a galerii. Zobrazení různých náhledů téže fotografie je řešeno pomocí odkazů na aktuální dokument s různou hodnotou parametru `dmode`. Díky předávání parametru `dmode` je také možné procházet fotografiemi se zachováním stejné velikosti náhledu. Definice vzhledu navigačního panelu stejně jako další formátování ovlivňující zobrazení sbírky jsou umístěny v konfiguračním souboru sbírky DKF. Použitá makra, konfigurační soubor sbírky DKF i zdrojové materiály a soubory s metadaty lze nalézt na přiloženém CD.

3.2.5 Zhodnocení výsledků

Jedním z úkolů této práce bylo vyzkoušet a popsat možnost použití digitální knihovny Greenstone pro správu sbírek fotografií. Podklady pro vytvoření sbírky byly získány z existujících sbírek spravovaných systémem DKF, který zároveň sloužil jako vzor základní požadované funkčnosti. V rámci vytváření této sbírky vznikly transformační šablony pro převod metadat do formátu používaného systémem Greenstone, plugin DKFPlug zajišťující správné začlenění zdrojových souborů spolu s přípravou podkladů nutných pro zajištění navigace při prohlížení sbírky a také soubory maker a definice prvků sloužících pro dosažení požadovaného vzhledu a fungování sbírky. Všechny uvedené výstupy lze použít při začleňování dalších

materiálů do sbírky nebo vytváření sbírek podobných. Postup v takovém případě bude mnohem jednodušší, neboť bude stačit pouze pomocí vytvořených šablon převést metadata a spolu se soubory obsahujícími fotografie je nahrát do adresáře pro import. Poté už stačí jen obvyklým způsobem spustit import a budování sbírky. DKFPlug spolu s existujícím konfiguračním souborem zajistí správné provázání dokumentů a vytvoření navigace – vše již bez zásahu uživatele.

Výsledná sbírka splňuje požadavky kladené v zadání a umožňuje snadnou správu i prezentaci fotografií. Digitální knihovna Greenstone také dovoluje použít aktivní prvky jako jsou JavaScript nebo applety v jazyce Java s jejichž pomocí by bylo možné zajistit i řadu dalších funkcí poskytovaných systémem DKF. Z těchto důvodů lze říci, že velká obecnost systému Greenstone není zásadní překážkou pro vytvoření specializovaných sbírek, které navíc mohou fungovat vedle sbírek méně vyhraněných, obsahujících například dokumenty různých formátů.

3.3 Sbírka dokumentů různých formátů – Paradox

Velkou předností digitální knihovny Greenstone je její schopnost uchovávat a zpřístupňovat dokumenty nejrůznějších formátů. Obvyklé formáty podporované knihovou sahají od textových dokumentů a internetových stránek vytvořených pomocí jazyka HTML až po obrázky, audio a video soubory. Při importování dokumentů je využito systému „zásuvných modulů“ (pluginů) napsaných v jazyce Perl, které dovolují definovat a upravovat způsob začleňování dokumentů nových formátů do sbírek digitální knihovny. Díky tomu je možné kromě specializovaných sbírek (jakou je například sbírka fotografií z DKF popsaná v minulé kapitole) vytvářet sbírky obsahující materiály odlišných formátů a zároveň zachovat jednotné uživatelské rozhraní.

Sbírka Paradox, která obsahuje dokumenty týkající se amatérského filmu Paradox, měla za úkol demonstrovat schopnost digitální knihovny Greenstone spravovat různorodé materiály se zachováním možnosti jejich snadného prohledávání, procházení a zobrazování. Zároveň slouží jako ukázka, jakým způsobem lze zasáhnout do uživatelského rozhraní a přizpůsobit jej potřebám tvůrců sbírek.

3.3.1 Zdrojové materiály a způsob jejich zpracování

Základem sbírky Paradox jsou scénáře a poznámky k filmu v podobě textu a HTML stránek, dále záběry z filmu a natáčení uložené jako obrázky ve formátu `jpg` a video soubory obsahující film samotný spolu s dalšími natočenými materiály. Obrázky a textové dokumenty jsou přímo součástí sbírky, videa byla z důvodů velikosti ponechána na původním umístění a jsou ze sbírky pouze odkazována. Aby uživatel získal dojem, že snímky jsou součástí sbírky, byly místo nich zařazeny zástupné soubory obsahující obrázky se záběry z filmů. Tyto obrázky zároveň slouží jako prostředek pro navázání potřebných metadat, která umožňují videa popsat a také mezi nimi vyhledávat.

Pro import dokumentů do sbírky byly využity pluginy, které jsou součástí instalace systému Greenstone – *TEXTPlug* pro textové dokumenty, *HTMLPlug* pro stránky HTML. Pro začlenění obrázků byl v rámci této diplomové práce vytvořen plugin *ImageImportPlug*. Greenstone sice poskytuje plugin pro zpracování obrázků (*ImagePlug*), ale ten je určen převážně pro systém Unix, neboť pro vytváření náhledů obrázků využívá další programy, které je nutné pro systém Windows doinstalovat. ImageImportPlug umožňuje začleňování obrázků s již vytvořenými náhledy a asociování externích metadat. Pomocí nastavení v konfiguračním souboru sbírky lze specifikovat, jakým způsobem bude plugin pracovat:

- V případě, že nejsou zadány žádné parametry, vytvoří ImageImportPlug pro každý grafický soubor zástupný dokument knihovny Greenstone. K tomuto zástupnému dokumentu poté asociuje samotný grafický soubor a dále veškerá externě přiřazená metadata spolu s metadaty určujícími interní adresu v rámci sbírky knihovny Greenstone (význam adres je popsán v podkapitole 3.2.3).
- Chování pluginu je možné ovlivnit zadáním tří parametrů, které dovolují určit ke každému grafickému souboru další asociované soubory (to je případ obrázku a jeho náhledů). První parametr s názvem `complex_document` sděluje pluginu, že má obrázky zpracovávat jako navzájem svázané dokumenty. Druhý parametr, `document_base`, pak určuje metadatový element, který odlišuje základní dokumenty (obrázky, které budou dále zpracovány) od ostatních (které budou buď asociovány s nějakým základním dokumentem, nebo nebudou vůbec zařazeny do sbírky). Posledním parametrem je `assoc_files` – obsahuje výčet metadatových elementů, jejichž obsah má být při zpracování brán jako popis umístění souboru s obrázkem, určeným pro navázání na základní dokument.

  Po zadání uvedených parametrů pak ImageImportPlug zpracovává jen ty soubory, které jsou určeny jako základní. Pro každý takový dokument provede stejné kroky, které jsou uvedeny při zpracování bez parametrů. Navíc projde všechny metadatové elementy zmíněné za argumentem `assoc_files` a pokusí se s dokumentem asociovat sobory, jejichž názvy tyto elementy obsahují. Chceme-li tedy do sbírky zahrnout obrázky svázané s náhledy, nastavíme v konfiguračním souboru sbírky uvedené parametry a u metadat příslušejících obrázkům doplníme do vybraných elementů odkazy na náhledy a atribut, který určí, že se jedná o základní dokument.

Na obrázku 13 je uveden příklad nastavení pluginu ImageImportPlug použitého v konfiguračním souboru sbírky Paradox. Záběry z filmu byly importovány spolu s náhledy – parametry určují, že základní dokumenty mají zadán element `dc.DocumentBase` a že název souboru obsahujícího náhled je uložen v elementu `dc.Thumbnail`.

```
plugin ImageImportPlug -complex_document
                       -document_base dc.DocumentBase
                       -assoc_files dc.Thumbnail
```

Obrázek 13: Parametry pluginu ImageImportPlug pro sbírku Paradox

S využitím prostředků zmíněných v této kapitole byly všechny zdrojové dokumenty zpracovány a začleněny do sbírky. Kromě metadat popisujících dokumenty samotné byla přidána i metadata umožňující vytvoření tří hierarchických klasifikátorů. Každý z nich nabízí strukturovaný náhled na jeden druh dokumentů obsažených ve sbírce: textové dokumenty, obrázky a videa. Spolu se seznamem všech dokumentů setříděných podle abecedy a možností vyhledávání tak sbírka Paradox poskytuje uživateli dostatečné spektrum různých přístupů k nabízeným materiálům.

3.3.2 Úprava uživatelského rozhraní

Systém pro správu digitálních knihoven Greenstone je do značné míry konfigurovatelný ve všech ohledech. Platí to jak o možnosti ovlivnit fungování knihovny samotné – od úpravy

konfiguračního souboru přes vytváření a úpravu pluginů až po zásah do samotného zdrojového kódu knihovny – tak i pro změnu podoby uživatelského rozhraní. Základním prostředkem pro přizpůsobení prezentace obsahu sbírek představám tvůrců a potřebám uživatelů jsou formátovací řetězce uložené v konfiguračním souboru každé sbírky. S jejich pomocí lze snadno upravit způsob jakým budou zobrazovány výsledky vyhledávání a klasifikátory, ale také způsob prezentace samotných dokumentů. Formátovací řetězce se skládají z textu, elementů jazyka HTML, podmíněných výrazů, názvů maker, systémových proměnných a odkazů na atributy dokumentů (metadata). Právě využití podmíněných výrazů, maker a atributů dokumentů dovoluje napsat obecné formátovací řetězce, které se jako šablony používají při vytváření podoby konkrétní stránky. V případě, že uživatel není spokojen se standardním zobrazením některých prvků, může je snadno změnit předefinováním formátovacích řetězců v konfiguračním souboru. Na následujícím obrázku je zobrazen příklad jednoduchého formátování seznamu dokumentů používaného klasifikátory:

```
Format VList {
  <td>[link]
    {If}{[dc.Image],
    <img
      src='_httpcollection_/index/assoc/[assocfilepath]/[dc.Image]'>,
    [icon]}[/link]
  </td>
  <td>
    {Or}{[dc.Title],[Title],Title is unknown}
  </td>
}
```

Obrázek 14: Ukázka formátovacího řetězce

Uvedený formátovací řetězec určuje jakým způsobem budou uživateli předkládány informace o každé položce seznamu. Údaje budou rozděleny do dvou buněk tabulky (elementy `<td>`). Pokud bude u dokumentu definován atribut `dc.Image`, zobrazí se v první buňce obrázek odkazovaný tímto elementem. V opačném případě se zobrazí zástupná ikona definovaná systémem Greenstone. Obrázek i ikona budou zároveň sloužit jako odkaz na daný dokument (systémová proměnná `[link]` zastupuje odkaz na umístění aktuálního dokumentu ve sbírce). Druhá buňka bude zobrazovat buď obsah elementu `dc.Title`, elementu `Title` a nebo se do ní vypíše řetězec určující, že název není znám.

Podobné, avšak složitější formátovací řetězce jsou použity i ve sbírce Paradox. Na obrázku 15 je uveden příklad hierarchického klasifikátoru určeného pro procházení video souborů. Podobně jako v příkladu uvedeném na obrázku 14 je i zde pro vypisování seznamu dokumentů použito tabulky, v jejíž první buňce se nachází náhled (pokud je k dispozici) a druhá buňka zobrazuje důležité informace (metadata). Detailnější popis způsobů formátování lze nalézt v kapitole „Makra“ (viz 3.6) nebo v literatuře (viz [3]).

Formátovací řetězce jsou ovšem prostředkem, který dovoluje zasahovat pouze do způsobu zobrazování obsahu sbírek a nikoliv do celkové podoby rozhraní digitální knihovny. Systém Greenstone je ale koncipován natolik obecně, že je možné se znalostí jazyka maker zasáhnout i do zobrazení stránek a ovládacích prvků celé knihovny. Naprostá většina stránek je vytvářena až při požadavku na jejich zobrazení. Podobně jako u formátovacích řetězců to dovoluje využití metasymbolů zastupujících makra a podmíněné větvení. Každá ze stránek je popsána pouze

šablonou, která je naplněna až na základě konkrétních nastavení prostředí (jazyk, způsob prohlížení). Protože tyto šablony jsou, stejně jako definice jednotlivých maker, uloženy v textových souborech, je možné je snadno prohlížet a editovat nebo vytvářet nové, které překryjí ty původní. Orientaci v makrech však ztěžuje fakt, že mohou být navzájem vnořována a při vytváření konkrétní podoby stránky jich pro zobrazení jednoho prvku může být expandována celá řada. Některá makra navíc slouží pouze pro učení formátování, zatímco jiná mohou obsahovat odkazy na konkrétní soubory (například ikony nebo obrázky).

Jedním z cílů sbírky Paradox bylo vyzkoušet a demonstrovat možnost úpravy uživatelského rozhraní na úrovni sbírky. Kromě úpravy formátovacích řetězců popisujících způsob prezentace obsahu byly provedeny také zásahy do způsobu zobrazení hlavičky a hlavního navigačního panelu. Hlavičkou je označována oblast, která obsahuje soubor ikon a funkčních prvků sloužících pro orientaci a navigaci v rámci sbírky. Lze ji rozdělit na čtyři oblasti:

- ovládací prvky knihovny (tlačítka `home`, `help` a `preference`)
- název sbírky
- název aktuálně zobrazené sekce
- navigační panel

Na obrázku 15 je uveden snímek sbírky Paradox s otevřenou stránkou pro procházení video souborů pomocí hierarchického klasifikátoru:

Obrázek 15: Sbírka Paradox

Srovnáním hlavičky stránky s ukázkou prezentovanou v teoretické části této práce (viz obrázek 3) je zřejmé, že došlo k určitým změnám. Ovládací prvky knihovny (tlačítka `home`, `help` a `preference`) byly ponechány beze změny, aby se předešlo zmatení uživatele. Také obrázek s logem a nápisem Paradox byl na místě názvu sbírky zobrazen standardním způsobem – zadáním hodnoty parametru `iconcollection` v konfiguračním souboru sbírky. Změny byly provedeny až ve způsobu zobrazení navigačního panelu a ikon určujících aktuálně prohlíženou sekci.

Položky nabízené v navigačním panelu přímo odpovídají klasifikátorům vytvářeným na základě nastavení v konfiguračním souboru. Pro každý funkční klasifikátor (tedy takový, který byl správně vytvořen a zobrazuje neprázdný náhled na dokumenty ve sbírce) existuje na navigačním panelu jedno tlačítko, které slouží k jeho zobrazení. U každé sbírky je navíc k dispozici možnost vyhledávání, které je zpřístupňováno pomocí tlačítka hledej (`search`),

umísťovaného vždy na levou stranu navigačního panelu. Pokud tedy například v konfiguračním souboru uvedeme, že chceme klasifikovat dokumenty podle názvu, objeví se při prohlížení vytvořené sbírky na navigačním panelu tlačítko „názvy a-z“ odkazující požadovaný klasifikátor. Velká část sbírek používá podobné klasifikátory – uvedený seznam názvů je nejlepším příkladem. Greenstone má pro tyto často používané klasifikátory vytvořeny soubory obrázků, které jsou použity pro zobrazení tlačítka na navigačním panelu a pro popis příslušné sekce. Pokud se vrátíme k uvedenému seznamu názvů dokumentů – na navigačním panelu se místo pouhého textu, který je používán v obecném případě, objeví pro zpřístupnění klasifikátoru názvů dokumentů obrázek znázorňující tlačítko. Po stisknutí tlačítka se jednak zobrazí požadovaný klasifikátor a také se patřičně změní obrázky tlačítka a aktuálně prohlížené sekce.

Pokud chceme určit vlastní podobu zobrazovaných tlačítek, máme dvě možnosti. Buď nadefinovat řadu maker pro zobrazení nového tlačítka a navázat je na použití určitého klasifikátoru nebo použít tlačítka stávající a změnit obrázky, které jsou jim standardně přiřazeny. U sbírky Paradox byl zvolen druhý přístup, neboť dovoluje rychlé dosažení požadovaného výsledku. Definice většiny obrázků, které byly v tomto případě předmětem zájmu, byly umístěny v balících maker `Global`, `Query`, `Document` a `Preferences` (více o makrech a významu jejich členění do balíků viz kapitola 3.6). Pro změnu zobrazení každého tlačítka a jemu příslušejícího popisu sekce bylo nutné vytvořit čtyři obrázky: běžnou podobu tlačítka, zvýrazněné tlačítko (když se nad ním nachází kurzor myši), vybrané tlačítko (je-li zobrazena sekce tlačítku příslušející) a konečně obrázek, který se zobrazuje nad navigačním panelem a indikuje aktuálně prohlíženou sekci.

```
_httpiconttitlgr_ [l=cs,c=paradox] {_httpprefix_/collect/paradox/images/ttitlescsgr.gif}
_httpiconttitlon_ [l=cs,c=paradox] {_httpprefix_/collect/paradox/images/ttitlescson.gif}
_httpiconttitlof_ [l=cs,c=paradox] {_httpprefix_/collect/paradox/images/ttitlescsof.gif}
```

Obrázek 16: Předefinování vzhledu tlačítek

Na obrázku 16 je ukázána část souboru s makry vytvořeného pro potřeby změny vzhledu rozhraní sbírky Paradox. Konkrétně se jedná o změnu obrázků pro tlačítko názvy. Na začátku každého řádku je uveden název makra, parametry v hranatých závorkách určují omezení na jeho použití (`[l=cs, c=paradox]` určuje, že makro má být použito pro sbírku Paradox zobrazenou v českém jazyce) a ve složených závorkách je uvedena cesta k novému obrázku. Podobným způsobem je předefinován vzhled všech dalších tlačítek, stejně jako obrázků sekcí a podkladu navigačního panelu. Všechna makra pro sbírku Paradox jsou umístěna v souboru `paradox.dm`, který je uložen na CD přiloženém k této práci. Popis postupu vedoucího ke změně vzhledu sbírky je také možné nalézt v dokumentu vytvořeném jedním z uživatelů systému Greenstone (viz [51]).

3.3.3 Zhodnocení výsledků

Úkolem sbírky Paradox bylo demonstrovat schopnost digitální knihovny Greenstone vytvářet a zpřístupňovat sbírky obsahující materiály různých formátů. Do sbírky byly začleněny textové dokumenty, internetové stránky, obrázky a video soubory. Skutečnost, že videa nejsou přímou součástí sbírky, neovlivňuje schopnost knihovny zahrnout je do vyhledávání a zobrazit o nich potřebné údaje. V případě potřeby má uživatel možnost pomocí odkazu video stáhnout z původního umístění mimo digitální knihovnu. Pro zpřístupnění dokumentů jsou nabízeny čtyři přístupy: vyhledávání, výběr ze seznamu všech dokumentů (nezávisle na jejich formátu)

uspořádaných podle názvů a hierarchické klasifikátory zpřístupňující zvlášť různé druhy dokumentů (text, obrázky a video).

Druhým úkolem sbírky Paradox bylo vyzkoušet a demonstrovat možnost změny vzhledu uživatelského rozhraní. V předchozích kapitolách byly stručně popsány principy použité pro zobrazování stránek digitální knihovny a také na konkrétních příkladech ukázány zásahy provedené při úpravě vzhledu sbírky Paradox.

Dosažené výsledky dokazují, že digitální knihovna Greenstone poskytuje dostatečné prostředky nejen pro správu různorodých materiálů, ale i pro jejich vhodné zpřístupnění uživateli. Možnost snadného ovlivnění vzhledu dokumentů a klasifikátorů pomocí formátovacích řetězců uložených v konfiguračních souborech sbírek, ale také možnost zásahu do flexibilního způsobu generování rozhraní celé knihovny, dovoluje využívat cenné služby digitální knihovny a zároveň upravit její vzhled podle konkrétních potřeb.

3.4 Zpřístupnění systému Greenstone českým uživatelům

Digitální knihovna Greenstone se díky své otevřenosti a snadné dostupnosti stává široce používaným nástrojem pro správu sbírek digitálních dokumentů. Spolupráce s organizacemi OSN a UNESCO při šíření důležitých poznatků do rozvojových oblastí stejně jako požadavky uživatelů z celého světa udržují systém ve stavu stále aktivního vývoje, přičemž důraz je kladen na zachování zpětné kompatibility. Dostupnost zdrojových kódů a možnost snadné úpravy fungování digitální knihovny také dovoluje přizpůsobovat její služby i velice nestandardním potřebám. Na vývoji jádra a jeho úpravách se podílí nejen tým odborníků z univerzity Waikato na Novém Zélandu, ale také velký počet uživatelů jak z odborné, tak i laické veřejnosti. Důležitým kanálem pro výměnu poznatků, kladení otázek a konzultaci připomínek k systému Greenstone tvoří především dvě e-mailové konference. První z nich je určena pro uživatele systému, druhá pro jeho vývojáře – tedy autory a správce sbírek (archiv konferencí viz [19]). Zvláště archiv zpráv z konference vývojářů a dotazy pokládané na této konferenci byly v některých případech důležitým zdrojem informací, na jejichž základě byly vytvářeny sbírky popsané v kapitolách této práce a získávány poznatky o fungování knihovny Greenstone. Díky uvedené konferenci jsem měl možnost dostat se do kontaktu se samotnými spoluautory systému a získat od nich cenné informace. Při komunikaci s nimi se také ukázala další z velkých výhod digitální knihovny Greenstone a tou je značná ochota jak autorů tak i uživatelů pomoci při jejím používání. Noví uživatelé systému a zájemci o něj by měli kromě vydávaných příruček a článků na internetu hledat odpovědi na své otázky také v uvedených archivech (v současné době už archivy obsahují i dotazy a poznámky, kterými jsem do konferencí přispěl při řešení této diplomové práce).

Jedním z cílů této práce bylo přiblížit digitální knihovnu Greenstone českým uživatelům. Kromě výsledku v podobě práce samotné byly provedeny i další kroky směřující k usnadnění použití systému Greenstone v českém prostředí. Podkapitola „Lokalizace uživatelského rozhraní“ popisuje způsob, jakým Greenstone řeší požadavky na použití knihovny v různých zemích a také zmiňuje překlad uživatelského rozhraní do češtiny provedený v rámci vypracování této práce. V podkapitole „Stemming“ je vysvětlen význam stemmingu při vytváření a používání sbírek digitální knihovny a uveden postup potřebný pro přizpůsobení stemmingu jiným jazykům. Poslední podkapitola – „Dokumenty týkající se systému Greenstone“ – se zabývá příručkou pro české uživatele, která byla vytvořena v rámci této diplomové práce.

3.4.1 Lokalizace uživatelského rozhraní

V teoretické části této práce bylo stručně zmíněno, že změny jazyka používaného v rámci uživatelského rozhraní se dosahuje využitím maker (viz kapitola 2.2.4). Makra obecně slouží pro definování vzhledu většiny stránek knihovny prezentovaných uživateli. Kromě formátování zobrazení obsahu sbírek zajišťují makra také správné vypisování textových řetězců sloužících pro komunikaci s uživatelem a zobrazování patřičných obrázků (příklad je uveden v popisu sbírky Paradox). Změny jazyka rozhraní je dosahováno předefinováním standardních maker spolu s určením podmínek jejich použití (v tomto případě se jedná o jazyk, zvolený v nastavení digitální knihovny).

Na uvedeném principu předefinování fungují i jazykové balíky, které lze stáhnout z oficiálních stránek knihovny Greenstone (viz [20]). Každý balík zahrnuje jednak nový soubor obsahující makra pro daný jazyk a jednak „lokalizované" obrázky. Ty obsahují například tlačítka navigačního panelu s přeloženými nápisy – místo původního obrázku pro tlačítko s nápisem „titles a-z" bude český balík obsahovat nový obrázek s nápisem „názvy a-z" a příslušné makro, které zajistí vykreslení tohoto obrázku v rámci českého rozhraní. Greenstone při zobrazování rozhraní vždy prochází seznamem souborů maker, který je uložený v hlavním konfiguračním souboru knihovny (soubor `main.cfg` v adresáři `etc`) a hledá makra vyhovující aktuálnímu nastavení knihovny a zobrazované stránce. Přednost mají soubory maker uvedené v seznamu později – díky tomu přidání názvu nového balíku na konec seznamu zajistí jeho přednostní použití. Mezi jednotlivými makry se pak vybere to, které nejvíce odpovídá daným podmínkám – například makro specifické pro určitou sbírku zobrazenou v konkrétním jazyce bude mít při splnění podmínek jazyka a sbírky přednost před makrem stejného názvu bez dalšího omezení použití. V případě, že není specifické makro nalezeno, prohledávají se ostatní makra téhož názvu, která splňují alespoň část podmínek.

Digitální knihovna Greenstone rozděluje míru lokalizace (*internationalizing*) do pěti skupin (viz [20], stránka *Internationalizing Greenstone*):

1) **Klíčové jazyky** (*core languages*) – pro tyto jazyky je vytvořena kompletní lokalizace rozhraní, dokumentace, příruček i ukázkových sbírek. Lokalizace pro jazyky této skupiny jsou vytvářeny ve spolupráci s organizací UNESCO. Mezi klíčové jazyky patří angličtina, francouzština, španělština a ruština.
2) **Plný překlad** (*full translation*) – překlad všech součásti jako u klíčových jazyků. V současné době existuje pouze pro kazaštinu.
3) **Udržované překlady rozhraní** *(maintained interface-only translation)* – existující překlad rozhraní a správce, určený pro jeho vytvoření a aktualizaci (nové verze Greenstone mohou obsahovat nová makra nebo měnit význam těch starých). Do této kategorie patří řada jazyků od arabštiny přes češtinu až po vietnamštinu.
4) **Neudržované překlady rozhraní** (*unmaintained interface-only translation*) – existující překlady klíčových prvků rozhraní, které ovšem nemají správce a proto mohou být zastaralé.
5) **Ve vývoji** (*in progress)* – rozhraní, na nichž se teprve pracuje a neexistuje zatím použitelná verze.

Pro překlady rozhraní se používají dva přístupy – on-line nástroj nazvaný *Translator* umožňuje procházet jednotlivá makra a doplňovat jejich lokalizované verze. Pro vytváření

nových překladů rozhraní nebo pro větší zásahy do již existujících se používá úpravy seznamů maker uložených v souborech formátu `excel`.

Prvotní lokalizace rozhraní do češtiny byla pro starší verze provedena panem Romanem Chýlou, studentem ÚISK FF UK Praha. S jeho pomocí a s pomocí pana Michaela Dewsnipa z univerzity Waikato, který má lokalizace rozhraní na starosti, jsem se stal oficiálním správcem překladu rozhraní pro češtinu (viz [20], stránka *Internationalizing Greenstone*, tabulka přehledu udržovaných překladů rozhraní). Jak již bylo zmíněno, čeština patří do třetí kategorie uvedené v seznamu míry lokalizace rozhraní. Jelikož původní překlad rozhraní pro češtinu nebyl téměř rok udržován, dostal jsem na starosti úpravu a doplnění seznamů maker uložených ve formátu `excel` (soubory se seznamy i výslednými přeloženými soubory českého rozhraní jsou uloženy na přiloženém CD). Přeložená makra v době dokončení této práce odpovídala aktuálně vydané verzi 2.50 systému Greenstone. Jakmile bude provedena konverze do formátu maker používaných systémem Greenstone a budou vytvořeny příslušné obrázky, stane se nový překlad součástí jazykového balíku pro češtinu, který je ke stažení na oficiálních stránkách digitální knihovny Greenstone.

3.4.2 Stemming

Pro rychlé získání dokumentu obsahujícího požadovaný textový řetězec používá Greenstone fulltextového vyhledávání. Aby nebylo nutné po zadání každého dotazu prohledávat všechny dokumenty sbírky, vytváří Greenstone při budování každé sbírky speciální struktury – indexy. Ty slouží jako „rejstříky“, které obecně obsahují odkazy na všechna slova každého dokumentu ve sbírce. Při hledání se pak prochází pouze indexy, na jejichž základě se určí všechny dokumenty, které obsahují některá (nebo všechna) slova dotazu.

Stemming je metoda, která ke každému slovu umožňuje určit jeho kořen. Uživatel digitální knihovny (stejně jako například internetového vyhledávače) často nepátrá po dokumentu, který obsahuje přesnou podobu hledaného výrazu, ale zajímají jej všechny dokumenty, které se k dotazu nějakým způsobem vztahují. S použitím stemmingu lze s omezením přesnosti (*precision*) zvýšit ohlas (*recall*) – tedy získat více dokumentů, které mohou odpovídat zadanému dotazu za cenu snížení přesnosti vyhledávání (ve výsledku se mohou objevit i s dotazem nesouvisející dokumenty). Například výsledkem hledání řetězce „africká budova“ s využitím stemmingu mohou být kromě dokumentů obsahujících přímo zadaný řetězec také dokumenty obsahující spojení „africké budovy“ – u všech slov zadaného dotazu je nejprve nalezen jejich kořen a poté jsou hledány všechny dokumenty obsahující slova s těmito kořeny.

Použití stemmingu v digitálních knihovnách kromě ovlivnění výsledků vyhledávání vede také ke zmenšení velikosti fulltextových indexů. Místo toho, aby se v nich uchovávaly odkazy na všechna slova, obsahují indexy pouze záznamy o kořenech slov spolu s odkazy na slova s těmito kořeny. Stemming tak vlastně provádí obecnější rozklad na množině slov a díky tomu uchovává informace o méně kategoriích. Digitální knihovna Greenstone vytváří při budování sbírek celkem tři indexy:

1) index všech slov bez ohledu na velikost znaků
2) index kořenů slov (využitím stemmingu) se zachování velikosti znaků
3) index kořenů slov bez ohledu na velikost znaků

Při hledání ve sbírkách je použit ten index, který odpovídá aktuálním nastavením (použití stemmingu lze stejně jako zohlednění velikosti znaků určit v nastavení digitální knihovny).

První index slouží pro přesná hledání zadaných výrazů. Druhý a třetí index pak umožňují použití stemmingu.

Nezbytným předpokladem použití stemmingu je správně fungující nástroj – *stemmer* – který je schopen pro konkrétní jazyk určovat kořeny slov. Stemmery jsou pro různé jazyky odlišné, neboť musí vyhovovat různým pravidlům utváření a skloňování slov. Greenstone zatím nabízí funkční stemming pouze pro angličtinu. V rámci cílů této práce bylo jedním z úkolů zjistit, jakým způsobem je možné upravit systém Greenstone tak, aby dovoloval použití stemmingu i pro češtinu.

3.4.2.1 Postup pro přidání nového stemmeru

Tato kapitola obsahuje technické detaily zprovoznění nového stemmeru pro použití v knihovně Greenstone. Spolu se stručným popisem, který jsem zaslal na e-mailovou konferenci vývojářů systému, se v současné době jedná o jediný dostupný návod, jak lze začlenit stemming pro nové jazyky.

V teoretické části této práce bylo uvedeno, že Greenstone dovoluje využívat dvou metod vytváření indexů – MG a MGPP (viz kapitola 2.2.3.2). V rámci zdrojových kódů knihovny odpovídá tomuto dělení také organizace jednotlivých souborů – zdrojové soubory pro MG jsou umístěny v podadresáři `packages/mg/` a pro MGPP v podadresáři `src/mgpp/`. Pro oba systémy jsou pro stemming použity shodné zdrojové kódy a proto budou odkazy dále uváděny jen pro systém MG. Pro zprovoznění nového stemmeru je třeba provést následující kroky:

1) **Vytvoření nebo získání zdrojových souborů stemmeru** – nejprve je nutné zajistit zdrojové kódy stemmeru v jazyce C/C++. Požadavek na použitý programovací jazyk vyplývá z faktu, že celé jádro systému Greenstone je naprogramováno v C++. Stemmer musí poskytovat veřejnou (*public*) funkci, která používá jeden vstupní/výstupní parametr. Vstupem bude vždy slovo, k němuž má být určen kořen, výstupem pak kořen tohoto slova. Hlavička funkce může mít podobu:

   ```
   void simpleczechstem (unsigned char *word);
   ```

   Pro představu se lze podívat na soubor `lovinstem.c` používaný pro angličtinu (je umístěn v podadresáři `packages/mg/lib` adresáře se zdrojovými kódy).

2) **Upravení zdrojového souboru stemmer.c** – v tomto zdrojovém souboru (je umístěn v adresáři `packages/mg/src/text`) se nachází dvě důležité funkce, které jsou volány přímo jádrem knihovny Greenstone při požadavku na zjištění kořenu slova. V rámci jedné z těchto funkcí se pak na základě parametrů prostředí zvolí, jaký stemming (tj. pro jaký jazyk) má být použit. Jedná se o funkce:

   - `int stemmernumber (u_char *stemmerdescription)` – na základě parametrů předaných z příkazové řádky určí interní identifikátor stemmeru, který má být použit. V této funkci se specifikuje, jaký parametr (textový řetězec) přísluší novému stemmeru.

- `void stemmer (int method, int stemmer, u_char *word)` – funkce volaná pro určení kořenu slova. Na základě parametru stemmer (odpovídá identifikátoru, který vrací první funkce) se použije příslušný stemmer.

Do zdrojového souboru `stemmer.c` je třeba přidat nový interní identifikátor (pro námi přidávaný stemmer) a volání nového stemmeru ve funkci `stemmer`.

3) **Zkompilování kódu** – na závěr je potřeba znovu zkompilovat zdrojové soubory knihovny Greenstone. Výsledné soubory, které dovolí volání nového stemmeru, jsou `mg_passes.exe` a `mg_stem_idx.exe` (pro systém Windows umístěné v adresáři `bin/windows`). Pokud zmíněnými soubory přepíšeme ty již existující, které se nachází v adresáři instalace systému Greenstone, můžeme začít námi přidaný stemming používat.

4) **Úprava skriptů používaných při budování knihovny** – posledním krokem, který je potřeba udělat, je změnit používání výchozího stemmeru. Při budování sbírek je kromě jiných volán také skript `mgbuilder.pl`. Právě tento skript má na starosti vytváření indexů a dalších důležitých struktur. V rámci tohoto skriptů je nutné do všech volání programu `mg_passes.exe` přidat parametr:

```
-a nazev
```

Název se musí shodovat s hodnotou, pro níž jsme v bodu 2) určili identifikátor. Ve standardním případě se programu `mg_passes.exe` nepředává parametr žádný a proto bývá použit výchozí stemmer pro angličtinu. Přidáním uvedeného parametru dosáhneme toho, že indexy pro všechny další sbírky budou vytvářeny s použitím námi přidaného stemmeru.

Přidání stemmingu pro indexační systém MGPP probíhá až na umístění a názvy jednotlivých souborů stejně. Zkompilováním upravených zdrojových souborů získáme programy `mgpp_passes.exe` a `mgpp_stem_idx.exe`. Parametr uvedený v kroku 4 je potřeba pro MGPP přidat ke všem voláním `mgpp_passes.exe` ve skriptu `mgppbuilder.pl`.

Poznatky uvedené v této kapitole byly získány na základě studia zdrojových kódů systému Greenstone. Stručné informace o umístění souborů vztahujících se ke stemmingu poskytl e-mailem pan John R. McPerson z univerzity Waikato. Při testování správnosti uvedeného postupu byl vytvořen a uveden do provozu jednoduchý program simulující stemmer. Ověřením správnosti uvedeného postupu může být i jeho potvrzení spoluautory systému Greenstone.

Pro zavedení funkčního stemmingu pro češtinu je možné použít například volně dostupný soubor nástrojů *Ispell* (viz [18]). Ačkoliv problém hledání kořenů slov neřeší Ispell způsobem vhodným pro češtinu, dostupnost zdrojových kódů a existence slovníků dovoluje jeho využití téměř okamžitě, alespoň do doby, než bude vytvořen lépe vyhovující nástroj.

3.4.3 Dokumenty týkající se systému Greenstone

Systém pro správu digitálních knihoven Greenstone je využíván v různých zemích celého světa, nemalou měrou také díky tomu, že je oficiálním prostředkem pro šíření informací v rámci programů organizací UNESCO a OSN. Přestože existuje řada dokumentů popisujících

Greenstone z různých pohledů a zaměřujících se na konkrétní problémy při jeho používání, žádný z nich není v českém jazyce. Navíc ani oficiální příručky vydávané autory systému neobsahují některé informace důležité pro nové uživatele.

Cílem této diplomové práce bylo poskytnout důležité informace o systému Greenstone českému uživateli spolu s poznatky získanými při jejím používání. Součástí zadání také bylo vystavit výslednou práci v elektronické podobě jako součást sbírky digitální knihovny Greenstone. Již v průběhu řešení jednotlivých úkolů spojených se zadáním bylo ale zřejmé, že požadavky na rozsah a formu diplomové práce neumožňují shrnout všechny získané poznatky do jednoho dokumentu. Proto byla současně s objevováním možností knihovny Greenstone vytvářena řada dokumentů, zachycujících důležité informace formou přístupnou i pro širší veřejnost. Všechny tyto dokumenty byly na závěr sloučeny do jediného, který na 40 stranách poskytuje základní poznatky o fungování digitální knihovny a jejích částí. Zároveň obsahuje kapitoly, popisující nástroje určené pro správu sbírek v rámci systému Greenstone – Collector a Librarian. Zvláště v případě nástroje Librarian, který poskytuje snadno použitelné uživatelské rozhraní k většině nabízených funkcí, neexistovala dlouhou dobu oficiální příručka pro jeho používání. Ta byla vydána až necelý měsíc před dokončením této práce.

Sbírka „Dokumenty o systému Greenstone“ obsahuje nejen kopii této diplomové práce, ale také příručku pro budoucí české uživatele. Shrnutí poznatků o nástroji Librarian a stručný návod na jeho používání je i součástí příloh (viz Příloha II).

3.5 Pluginy a jejich tvorba

Ačkoliv je jádro systému pro digitální knihovnu Greenstone vytvořeno v jazyce C++ a nelze do něj téměř zasahovat (výjimkou je možnost zkompilování zdrojových kódů s případnými vlastními úpravami), potřebné flexibility a rozšiřitelnosti funkčnosti je dosahováno pomocí bloků kódu napsaných v jazyce Perl. Tyto bloky, nazývané moduly, jsou určeny jak pro zpracování dokumentů při importu a operacích s tím souvisejících, tak i při prohlížení hotových sbírek. Jejich velkou výhodou je možnost snadných úprav kódu bez nutnosti překladu. Součástí instalace je velké množství modulů, které se podílejí na chodu knihovny – většinu z nich by však uživatel neměl měnit pokud dobře nerozumí jejich funkci, aby nezpůsobil chybné chování systému Greenstone.

Zvláštní skupinou modulů jsou takzvané pluginy. Slouží jako nástroje pro předzpracování dokumentů ve fázi importu a dovolují vytvářet sbírky z dokumentů různých formátů. V rámci standardní instalace Greenstone je k dispozici celá řada vytvořených pluginů, které zpracovávají dokumenty s nejběžnějšími formáty (`html`, `txt`) stejně jako umožňují konvertovat některé rozšířené formáty (`pdf`, `word`) a získané údaje používat pro vytváření indexů. Občas se ale může stát, zvláště pokud se pracuje s málo rozšířenými formáty dat nebo je vyžadováno nestandardní zpracování určitých dokumentů, že nastane potřeba vytvořit plugin nový. Uživatelé nemusejí mít obavy při experimentování s tvorbou a úpravou pluginů a tvůrci systému je k tomu dokonce sami vybízejí. Špatně fungující plugin totiž nezpůsobí téměř žádnou škodu – v nejhorším případě nezpracuje požadované dokumenty a sbírka bude prázdná.

Pro orientaci v rámci zdrojových kódů pluginů, stejně jako pro jejich případnou úpravu, je třeba znát programovací jazyk Perl (viz např. [37]). Kromě základní syntaxe, která je velmi flexibilní a umožňuje člověku se znalostmi programování rychlé osvojení, je důležité znát význam a vytváření regulárních výrazů a práci s datovými strukturami (zvláště s poli a asociativními poli). Pro experimentování stejně jako pro vytvoření nového kódu je dobré

zkopírovat již existující plugin a jeho úpravami dosáhnout požadovaného chování ve fázi importu dokumentů.

3.5.1 Umístění a hierarchie pluginů

Všechny moduly lze nalézt v podadresáři `perllib` adresáře s instalací Greenstone. Adresář `perllib` obsahuje moduly poskytující různé funkce využívané při importu a zpracování sbírek – konverzi do různých kódování, extrahování metadat a podobně. V podadresáři `plugins` se pak nacházejí samotné pluginy. Každý z nich je uložen jako samostatný textový soubor obsahující kód v jazyce Perl.

Každý plugin musí z důvodů správného fungování v rámci importu nabízet určité funkce a daným způsobem reagovat na podněty zvenčí – přebírat parametry od skriptů řídících proces importu a poskytovat srozumitelné výstupy. Část kódu je tedy pro všechny pluginy stejná nebo značně podobná. Z těchto důvodů se tvůrci systému rozhodli zasadit pluginy do hierarchie podobné dědičnosti, ve které nové chování může být snadněji definováno úpravou chování předka, přičemž kód pro sdílené (stejné) zpracování není třeba opakovat.

Nejobecnějším předkem v hierarchii je plugin s názvem *BasPlug*. Nemá praktické využití při importu dokumentů, pouze definuje základní rozhraní pluginů (předávané argumenty) a funkce inicializující a řídící zpracování dokumentů. Odvozené pluginy pak přidávají vlastní argumenty a upravují nebo přidávají další funkce. Tyto kapitoly obsahují základní popis struktury pluginů a informace použitelné při tvorbě pluginů nových. Zájemci o detailnější komentář k již existujícím pluginům mohou potřebné informace nalézt v příručkách poskytovaných ke Greenstone (viz [3]).

3.5.2 Zpracování dokumentu při importu

Jediným účelem fáze importu a tedy i většiny pluginů je zpracovat soubory, které se mají stát základem sbírky, do formy dále použitelné ve fázi budování sbírky. Na vstupu jsou soubory různých formátů, případně opatřené metadaty uloženými ve zvláštních souborech.Výsledkem úspěšného importu pak jsou XML soubory obsahující jak metadata, tak i data samotných dokumentů a asociované soubory. Je-li například vstupem HTML stránka obsahující text a obrázky, výstupem importu bude XML soubor obsahující metadata (název dokumentu, datum vzniku) a upravený text spolu s obrázky, které budou ve formě asociovaných souborů uložené ve zvláštním adresáři.

Při importu jsou jednotlivé soubory uložené v adresářích pro import postupně předkládány pluginům uvedeným v konfiguračním souboru sbírky. Každý z pluginů v závislosti na příponě souboru a parametrech určených v konfiguračním souboru sbírky ověří, zda je soubor schopen zpracovat a podle toho jej buď zpracovávat začne, nebo jej předá dalšímu pluginu. Neexistuje-li plugin, který umožní určitý soubor importovat, nebo nezdaří-li se zpracování souboru, není soubor do výsledné sbírky zařazen.

Během samotného zpracování se pro každý soubor, který má být základem dokumentu uchovávaného knihovnou, vytvoří zástupný objekt. K tomuto objektu je přiřazen unikátní identifikátor (OID – *object identifier*), který jej jednoznačně určuje v rámci celé sbírky. Dále jsou k objektu připojeny všechny údaje uložené jak v rámci externích metadat, tak i v rámci metadat specifikovaných uvnitř souboru samotného. U některých souborů se dále provádí zpracování jejich obsahu – extrahují se metadata, formátuje se text nebo se obsah převádí do zcela jiného formátu (příkladem může být konverze obsahu `pdf` souborů do textové podoby).

Je také možné asociovat další soubory – ty budou přístupné v nezměněné podobě pomocí odkazu uloženého v rámci metadat dokumentu. Na závěr je zástupný objekt předán ke zpracování systému Greenstone, který na základě jeho atributů vytvoří soubor ve formátu XML a zkopíruje asociované soubory. Výsledkem úspěšného importu je tedy XML soubor popisující původní dokument a jeho obsah spolu s asociovanými soubory.

Uvedený postup lze demonstrovat na zpracování HTML stránky – soubor, v němž je definice stránky uložena, je postupně nabízen pro zpracování jednotlivým pluginům. V obecném případě se zpracování ujme plugin s názvem HTMLPlug, který je součástí standardní instalace knihovny Greenstone. Nejprve se vytvoří zástupný objekt a vygeneruje se pro něj unikátní identifikátor (OID). Dále se k objektu přiřadí metadata, která byla k souboru připojena externě pomocí výčtu atributů a jejich hodnot ve zvláštním XML souboru. V závislosti na parametrech se pak plugin může pokusit extrahovat metadata obsažená v souboru samotném – například název uvedený v hlavičce HTML souboru. Poté je zpracován samotný text souboru – formátovací elementy jsou upraveny pro uložení v XML, je změněna podoba interních a externích odkazů a údaje o umístění obrázků jsou v podobě metadat připojeny k objektu. Upravený text je na závěr přidán do zástupného objektu. Systém Greenstone poté na základě všech údajů uchovávaných zástupným objektem vytvoří XML soubor, ve kterém budou všechny informace uloženy. Zároveň k tomuto souboru do zvláštního adresáře nakopíruje všechny obrázky odkazované z původní stránky.

3.5.3 Základní struktura odvozeného pluginu

Konkrétní implementace jednotlivých pluginů je ve velké míře závislá na rozhodnutí jejich autorů. Většina pluginů však zachovává obecnou strukturu používanou v základním pluginu BasPlug. Znalost této struktury a významu důležitých bloků kódu značně usnadňuje orientaci v existujících pluginech a dovoluje snáze implementovat pluginy vlastní. Každý plugin obsahuje čtyři důležité bloky kódu:

- definice argumentů a jejich načítání
- vytvoření instance daného pluginu
- přijmutí dokumentu pro import
- zpracování obsahu dokumentu

Definice argumentů určuje, jaká nastavení jsou u daného pluginu k dispozici a jak ovlivňují jeho fungování – základním popisem se zabývá kapitola „Argumenty“. **Vytvoření instance** inicializuje plugin jako objekt pro zpracování souborů. **Přijmutí dokumentů** slouží pro rozhodování o tom, zda je daný plugin schopen určitý soubor zpracovat a vytváří instance dokumentů pro systém Greenstone. **Zpracování obsahu dokumentů** zahrnuje funkce upravující obsah dokumentů, extrahování metadat a další operace definované pro daný typ souboru. Popisem procesu inicializace pluginu a zpracování dokumentu se zabývá kapitola „Funkce read a process“. V následujících kapitolách budou také uvedeny nejdůležitější části kódu pluginů, jejichž znalost je klíčová pro vytváření vlastních modulů.

3.5.4 Argumenty

Argumenty (*arguments*) zahrnují parametry a k nim definované hodnoty. Tato nastavení ovlivňují způsob, jakým bude plugin zpracovávat importované soubory. Mezi argumenty

definované v základním pluginu BasPlug patří například určení přípon akceptovaných souborů, vstupní kódování nebo odkaz na zástupný obrázek. Jednotlivé parametry pluginů jsou uloženy v konfiguračním souboru sbírky a při spuštění importu předány pluginu (více o konfiguračním souboru lze najít v literatuře, viz [3]).

Definice argumentů jsou většinou umístěny na začátku kódu pluginu ve formě asociativního pole (`hash`). Jako ukázka je uvedena definice nastavení určujícího typy souborů, které mohou být pluginem zpracovány:

```
my $arguments =
    [ { 'name' => "process_exp",
        'desc' => "{BasPlug.process_exp}",
        'type' => "string",
        'deft' =>  &get_default_process_exp() }]
```

Obrázek 17: Ukázka definice argumentů pluginu

Většina argumentů je určena čtyřmi charakteristikami: jménem (`name`), popisem (`desc`), typem (`type`) a výchozí hodnotou (`deft`). **Jméno** slouží pro odlišení parametrů. Všechny znaky uvedené v nastavení pluginu v konfiguračním souboru za jménem argumentu se automaticky přiřazují tomuto argumentu. **Popis** slouží jako informaci pro uživatele; může obsahovat vysvětlení významu parametru či jiný komentář. **Typ** určuje jaké druhy hodnot mají být očekávány. Hodnotou tohoto parametru může být řetězec (`string`), číslo (`int`) nebo přepínač (`flag`). Přepínače slouží pouze pro zapnutí či vypnutí určité funkčnosti. Příkladem argumentu s parametrem flag může být nastavení „asociovat soubory“. **Výchozí nastavení** pak určuje hodnotu parametru, která bude použita, pokud uživatel nespecifikuje jinak. V příkladu na obrázků 17 je v definici výchozí hodnoty parametru uveden odkaz na funkci, jejíž návratovou hodnotou je řetězec. Protože uvedená funkce je často používána pro získávání akceptovaných souborů, je vhodné uvést ji také:

```
sub get_default_process_exp {
  my $self = shift (@_);
  return q^(?i)(\.jpe?g|\.gif|\.png|\.bmp|\.xbm|\.tif?f|\.dkf)$^;
}
```

Obrázek 18: Funkce definující akceptovatelné přípony

Návratová hodnota specifikuje regulární výraz, který bude sloužit pro porovnání se jménem souboru – v tomto případě slouží pro porovnání přípona souboru. V uvedeném příkladu bude plugin akceptovat většinu známých grafických formátů spolu se soubory s příponou `dkf` (tato funkce byla převzata z pluginu DKFPlug, jehož fungování je nastíněno v kapitole 3.2.3).

Podobným způsobem je možné vytvořit libovolný počet vlastních parametrů, na jejichž základě bude probíhat zpracování souborů. Jako inspiraci lze použít kterýkoliv z již vytvořených pluginů, uložených v dříve zmíněném adresáři `perllib/plugins`. Je ale potřeba mít na paměti, že řadu argumentů přebírá každý plugin od svých předků, přinejmenším od pluginu BasPlug. Pokud potřebujeme význam určitých nastavení změnit, je třeba předefinovat navázané chování. V opačném případě budou parametry zpracovány způsobem definovaným v předkovi.

3.5.5 Funkce read a process

Mezi nejdůležitější funkce každého pluginu patří funkce `read` a `process`. Jejich důležitost do značné míry závisí na tom, jak se uživatel rozhodne ke zpracování dokumentů přistupovat. Základní představa tvůrců systému je taková, že funkce `read` by měla u většiny pluginů zůstat původní – tedy zachovat funkčnost definovanou v pluginu BasPlug. Změny by měl tvůrce uplatňovat až ve funkci `process`, která je z funkce `read` volána.

Funkce `read` je spouštěna při každém předložení importovaného souboru pluginu. Obsahuje porovnání názvu souboru s akceptovanými příponami (viz obrázek 18 v předchozí kapitole) a přijetí či zamítnutí souboru. V případě přijetí je vytvořen zástupný objekt dokumentu, přiřazen identifikátor a metadata. Poté je řízení procesu importování předáno funkci `process` s potřebnými argumenty odkazujícími objekt dokumentu a další nastaveni. Veškeré další zpracování obsahu dokumentu a práce se zástupným objektem je dále v režii funkce `process`.

V případě, že potřebujeme zasáhnout pouze do způsobu zpracování samotného obsahu souboru (například upravit formátování textu HTML souborů), stačí předefinovat pouze funkci `process`. Struktura pluginů odvozená z pluginu BasPlug zajistí, že funkci `process` bude v pravý čas předáno řízení importu. V některých případech je však potřeba ovlivnit i způsob, jakým budou vytvářeny interní dokumenty zastupující jednotlivé soubory v rámci digitální knihovny Greenstone. Týká se to hlavně pluginů, které jsou určeny pro zpracování různorodých formátů dokumentů nebo které potřebují zasáhnout do procesu generování unikátních identifikátorů. V takovém případě je třeba předefinovat samotnou funkci `read` a další využití funkce `process` už závisí jen na vůli autora – místo ní může totiž být volána jakákoliv jiná funkce, která zajistí požadované zpracování obsahu souboru.

Závěr této kapitoly je věnován popisu nejdůležitějších bloků kódu funkce `read`. Jejich znalost do značné míry usnadňuje orientaci v kódu existujících pluginů a dovoluje sledovat klíčové kroky při importu dokumentů.

Na začátku funkce `read` jsou inicializovány proměnné pro uchovávání důležitých parametrů předaných pluginu. Především se jedná o odkaz na instanci pluginu a řídící data:

```
my $self = shift (@_);
my ($pluginfo,$base_dir,$file,$metadata,$processor,$maxdocs) = @_;
```

Důležitá je i podmínka, která rozhoduje o tom, zda má být předkládaný soubor zpracován tímto pluginem a nebo má být odmítnut. Název souboru je porovnáván se seznamem definovaným v rámci parametrů pluginu (viz příklad na obrázku 18 v kapitole 3.5.4):

```
if ($filename !~ /$self->{'process_exp'}/ || !-f $filename) {
  return undef; }
```

V případě, že předkládaný soubor je pluginem akceptován, dojde k vytvoření zástupného objektu pro budoucí dokument systému Greenstone. Zároveň je u dokumentu nastaven typ unikátního identifikátoru (`OIDtype`):

```
my $doc_obj = new doc ($filename, "indexed_doc");
$doc_obj->set_OIDtype ($processor->{'OIDtype'});
```

Greenstone dovoluje ke každému souboru definovat metadata externě. Tato metadata jsou ve fázi importu předávána pluginu současně se souborem, kterému příslušejí. Následující blok kódu ukazuje volání funkce pro zpracování takto získaných metadat a funkce `process`, sloužící pro zpracování samotného obsahu souboru. Na závěr je volána funkce pro extrakci interních metadat (například již zmíněného názvu umístěného v hlavičce `html` souborů):

```
$self->extra_metadata ($doc_obj, $doc_obj->get_top_section(),
                       $metadata);
return undef unless defined ($self->process (\$text, $pluginfo,
                        $base_dir, $file, $metadata, $doc_obj));
$self->auto_extract_metadata ($doc_obj);
```

Na závěr bývá dokumentu přiřazen jeho unikátní identifikátor (OID) a výsledek se předává k dalšímu zpracování systému Greenstone, který vytvoří XML soubor s popisem dokumentu:

```
$doc_obj->set_OID();
$processor->process($doc_obj);
```

Funkci pro generování unikátního identifikátoru (`set_OID()`) je také možné předat parametr. Bez parametru se identifikátor generuje standardním způsobem pomocí jednoznačného otisku souboru (*hash*). Uvedeme-li parametr, systém Greenstone použije jeho hodnotu jako identifikátor dokumentu.

3.6 Makra

Stejně jako v případě pluginů je i vzhled stránek ovlivňován pomocí uživatelem snadno editovatelných modulů umístěných odděleně od samotného jádra zajišťujícího chod knihovny. Podoba všech stránek je vytvářena až za běhu, co více, až při samotném zobrazení konkrétní stránky. Mechanismem který umožňuje tento způsob prezentace a zároveň jednoduchou editaci, je použití takzvaných maker. Makra jsou textové řetězce definovaného formátu, která zastupují bloky (převážně) HTML kódu, ovlivňujícího vzhled stránek, případně samotné zobrazované informace. Definice maker jsou umístěny v textových souborech. Aktuální nastavení prostředí (jazyk, způsob zobrazení, určitá sbírka) umožňuje měnit způsob expanze jednotlivých maker a tím dynamicky ovlivňovat vzhled stránek.

3.6.1 Balíky, umístění, rozdělení a fungování maker

Jednotlivá makra jsou logicky členěna do takzvaných balíků (*packages*), které odpovídají oblastem použití. Některé balíky obsahují makra ovlivňující vzhled určitých stránek, jiné zase zahrnují makra používaná v rámci celé knihovny. Názvy maker bývají často stejné a jen umístění do určitého balíku rozhoduje o tom, kdy bude které použito. Příkladem může být makro `content`, které obecně určuje vzhled stránky. Balíky jsou v rámci souborů obsahujících definice maker uvozeny pomocí klíčového slova `package` a názvu. Ačkoliv jeden soubor může obsahovat řadu různých balíků (v extrémním případě by mohla být dokonce všechna makra umístěna v souboru jednom), bývají většinou makra z organizačních důvodů rozdělena do více oddělených textových souborů. Ty jsou umístěny v podadresáři `macros` adresáře instalace knihovny Greenstone. Pro rozlišení jsou opatřeny příponou dm – například `about.dm` nebo `czech.dm`.

Samotná makra lze rozlišit podle dvou hledisek. Podle možnosti editace na **systémová** a **editovatelná** a podle vnitřní složitosti na **jednoduchá** a **složitá**. **Systémová** makra jsou definována a vyhodnocována v samotném jádru digitální knihovny a uživatel obyčejně nemá možnost je ovlivnit. Spíše než určení vzhledu stránek obsahují tato makra obdobu „systémových proměnných“ – odkazy na adresáře sbírek, odkazy na jednotlivé dokumenty a podobně. **Editovatelná** makra jsou všechna ostatní, tedy ta, která můžeme najít a podle potřeby měnit či přidávat. Druhé hledisko je subjektivnější a odpovídá tomu, jak „důležitá“ v hierarchii makra jsou. Již uvedené makro `content` je z tohoto pohledu složité – jednak zastupuje celou stránku a jeho neexistence tedy bude mít za důsledek zobrazení prázdné stránky a jednak je jeho definice složená kromě HTML elementů a textu také z volání dalších maker. **Jednoduchá** makra pak jsou taková, která sama o sobě slouží pouze jako prostředek pro ovlivnění způsobu zobrazení jednotlivých elementů – texty v různých jazycích nebo jednoduché bloky kódu.

Abychom zpřístupnili využívání maker v rámci prezentace digitální knihovny Greenstone a jejího obsahu, musíme v hlavním konfiguračním souboru knihovny (`etc/main.cfg`) přidat název souboru obsahujícího definice maker jako parametr nastavení `macrofiles`. Důležité je pořadí, ve kterém jsou názvy souborů uvedeny. Makra definovaná v těch později uvedených jsou upřednostňována. Za běhu je makro expandováno v závislosti na nastaveních prostředí a balíku, do nějž přísluší. Například při zobrazení úvodní stránky v češtině se hledá makro `content`, příslušející do balíku `home` definované pro parametr `[l=cs]`. Pokud toto makro není definováno pro češtinu, použije se standardní definice, která odpovídá anglickému jazyku. V rámci expanze obsahu stránky narazí systém na další makra, například `textaboutgreenstone`. I definice tohoto makra je hledána v balíku `home`. Pokud hlavní konfigurační soubor knihovny obsahuje odkaz na soubor maker `czech.dm`, bude pro expanzi makra použita definice:

```
_textaboutgreenstone_ [l=cs] {<p>}
```

Stejným způsobem probíhá zpracování všech maker. Je důležité mít na paměti, že pořadí souborů s definicemi maker určuje pořadí při vyhledávání definice makra. Popis základních balíků, způsobu vytváření nových maker a jejich využívání je rozebrán v následující kapitole.

3.6.2 Základní balíky a parametry prostředí

Makra jsou do skupin (balíků) rozdělována podle oblastí využití. Základní definice všech obecných maker různých balíků jsou umístěny v souboru base.dm. Většina maker je pak definována v dalších souborech, které jsou už specializovány jen na určité balíky.

Standardní balíky se většinou nacházejí v souborech stejných jmen – například balík `help` v souboru `help.dm`. V případě „jazykových“ maker, která jsou především zaměřena na překlady textů zobrazovaných v rámci uživatelského rozhraní knihovny, bývají definice všech maker různých balíků umístěny do jednoho souboru. Například soubor `czech.dm` obsahuje překlady všech důležitých textů rozhraní Greenstone do češtiny. Přehled a krátký popis jednotlivých balíků nabízí následující tabulka:

Název balíku	Popis
Global	Obecná makra používaná v rámci celé digitální knihovny.
home	Definice vzhledu domovské (úvodní) stránky knihovny. Lze zde například určit, jakým způsobem budou nabízeny obsažené sbírky,
help	Definice maker stránek nápovědy.
preference	Definice maker stránek nastavení digitální knihovny.
style	Makra ovlivňující styl prezentace stránek.
about	Používá se při zobrazení stránky o sbírce. Standardně na základě vypisuje obecné informace o sbírce a přístupu k dokumentům.
browse	Makra vztahující se ke stránkám obsahujícím klasifikátory. Určují formátování klasifikátorů.
query	Určení vzhledu stránky pro vyhledávání ve sbírce.
document	Definuje makra používaná při zobrazení dokumentů ve sbírek.

tabulka 2: přehled základních balíků maker

Již v minulé kapitole byl naznačen význam a použití parametrů prostředí při detailnější specifikaci definice makra. Základními parametry jsou jazyk, mód zobrazení a sbírka. Patřičnou úpravou konfiguračního souboru knihovny je také možné vytvořit parametry vlastní (viz kapitola 3.2.4.1). Obecné, dále neomezené makro má podobu:

```
_nazevmakra_ {definice obsahu}
```

Takto uvedené makro předefinuje všechna makra stejného názvu spadající do stejného balíku a bude uplatněno v případě, že se nenalezne žádné makro odpovídající dalším omezením. Často ale potřebujeme, například v závislosti na jazyku nebo sbírce, změnit podobu některých prvků. Můžeme použít jednoduchého testu parametrů prostředí, které určí, zda má být dané makro použito:

```
_mojemakro_ {Default macro}
_mojemakro_ [l=cs] {Makro pro český jazyk}
_mojemakro_ [l=cs,c=paradox] {Makro pro sbírku Paradox
                              zobrazenou v českém jazyce}
```

Pořadí uplatňování omezení je dáno nastavením parametru `macroprecedence`, jehož deklarace je uvedena v konfiguračním souboru knihovny (`etc/main.cfg`). Určuje, který parametr prostředí má přednost při uplatňování definice makra. Například nastavení:

```
macroprecedence c,v,l
```

určuje, že makra specifická pro sbírku (parametr `c`) mají přednost před makry pro zobrazení v textovém módu a makry pro určitý jazyk (parametry `v` a `l`).

Právě detailnější specifikace definic maker a mechanismus jejich předefinovávání nabízí silný nástroj pro úpravu vzhledu knihovny. Pokud například chceme změnit podobu úvodní stránky, je lepší vytvořit nový soubor s definicemi maker a přidat jej k nastavení v konfiguračním souboru digitální knihovny. Zároveň tak změníme vzhled stránky a současně zachováme původní definice maker pro případné pozdější znovupoužití. Mechanismus

předefinovávání maker je také uplatněn při překladech rozhraní Greenstone – do příslušného adresáře je pouze nahrán soubor obsahující definice maker pro daný jazyk a je upraven konfigurační soubor. Poté stačí obnovit nebo znovu otevřít stránky knihovny a rozhraní v novém jazyce bude okamžitě k dispozici.

3.6.3 Používání maker

Informace uvedené v předchozích kapitolách by měly stačit k tomu, aby bylo možno jak upravovat makra existující, tak i vytvářet vlastní. Celý postup je velice jednoduchý. Nejprve zvolíme, jakou část rozhraní chceme ovlivnit a podle toho určíme do jakého balíku bude nové makro patřit. Poté založíme v adresáři `macros` textový soubor (pokud budeme používat rozšířenou znakovou sadu, pak je třeba použít kódování UTF-8), do nějž vepíšeme hlavičku odkazující zvolený balík a můžeme začít definovat vlastní makra. Každé makro je uvozeno svým názvem ohraničeným podtržítky, omezeními kladenými na hodnoty parametrů prostředí a samotnou definicí makra, uzavřenou do složených závorek. Na závěr upravíme nastavení parametru `macrofiles` v konfiguračním souboru knihovny přidáním názvu právě vytvořeného souboru. Od této chvíle můžeme začít nová makra používat.

První způsob využití maker je, až na ovlivnění parametry prostředí, vcelku nezávislý na uživateli resp. tvůrci sbírky. Při zobrazení stránek je za běhu vygenerován jejich vzhled podle definic uvedených v patřičných souborech. Druhý způsob je používání maker při tvorbě formátovacích řetězců ovlivňujících prezentaci klasifikátorů a samotných dokumentů knihovny. Základní principy vytváření formátovacích řetězců jsou popsány v kapitole věnované sbírce materiálů o filmu Paradox (viz 3.3.2). Kromě dříve uvedených systémových maker (odkazujících například na adresář sbírky) můžeme nyní začít používat i makra vlastní. Pokud například potřebujeme u klasifikátoru zobrazujícího přehled dokumentů sbírky uvést u každého dokumentu vybrané údaje uvozené popiskem a chceme, aby tento popisek odpovídal jazyku rozhraní, můžeme použít makra. Vytvoříme si vlastní soubor maker umístěných do balíku `Global`, ve kterých budeme definovat překlady těchto popisků. Soubor může obsahovat například:

```
package Global
_txttitle_  {Title}
_txttitle_ [l=cs] {Název}
_txtuntitled_ {Untitled}
_txtuntitled_ [l=cs] {Bez názvu}
```

definující popisky uvozující název dokumentu a text pro neznámou hodnotu názvu. Takto definovaná makra potom můžeme ve formátovacím řetězci příslušného klasifikátoru snadno odkazovat a docílit tak popisku v tom jazyce, který uživatel zvolí při prohlížení sbírky:

```
<tr>
  <td align=\"right\" width=\"60\" valign=\"top\">
    <b>_txttitle_</b>
  </td>
  <td bgcolor=\"#FFFFFF\"> 
    {Or}{[dc.Title],_txtuntitled_}
  </td>
</tr>
```

Jazyk maker je koncipován tak, aby bylo možné jej snadno pochopit a ihned používat. Vzhled většiny stránek digitální knihovny Greenstone je vytvářen až při jejich zobrazování právě díky využití maker. Prohlížením souborů maker, které jsou součástí instalace, jejich editací a sledováním dopadu na podobu stránek, lze jazyk maker rychle ovládnout. Jeho využíváním je pak možné vytvářet sbírky a knihovny, jejichž vzhled a fungování bude odpovídat našim představám.

4 Závěr

Systém pro správu digitálních knihoven Greenstone patří hlavně díky úzké spolupráci s organizacemi OSN a UNESCO při šíření informací do rozvojových zemí k rychle se vyvíjejícím a perspektivním nástrojům pro správu sbírek digitálních materiálů. Velkou výhodou je jeho otevřenost a snadná dostupnost. Řada uživatelů z celého světa přispívá ke zdokonalování systému nejen připomínkami, ale také návrhy a úpravami kódu. Pro výměnu zkušeností mezi uživateli a tvůrci systému existují dvě e-mailové konference, které jsou nejen zdrojem cenných informací pro nové uživatele, ale umožňují také řešit problémy nové a konzultovat je se samotnými autory systému. Greenstone staví na použití silných a ověřených standardů a snadno dostupných a funkčních technologií. Velká obecnost uplatňovaná při návrhu i implementaci systému dovolují jeho použití nezávisle na obsahu nebo jazykových rozdílech. Vedle sbírek v anglickém jazyce existují sbírky materiálů arabských nebo dokonce čínských, samotné materiály pak jsou nejrůznějších formátů – od obyčejného textu přes internetové stránky a soubory textových procesorů až po obrázky, zvukové nahrávky nebo animace.

Ačkoliv je Greenstone stále více používán po celém světě, neexistoval dosud žádný ucelený popis v českém jazyce. Ani oficiální anglické manuály ale neobsahují shrnutí všech potřebných informací, které začínající uživatel potřebuje pro hlubší proniknutí do systému. Existuje řada zdrojů, které tyto informace obsahují, jejich nalezení však může zabrat mnoho času a nutí uživatele věnovat se spíše zjišťování postupu jak problémy řešit, než se zabývat jejich skutečným řešením.

Cílem této práce bylo prozkoumat systém Greenstone a přiblížit jej českým uživatelům. Současně s vysvětlením základních principů a mechanismů fungování digitální knihovny bylo úkolem prezentovat praktické zkušenosti získané při jejím používání a poznatky shromážděné o důležitých problémech z různých zdrojů; součástí práce bylo i navrhnout a implementovat pomocné nástroje pro efektivnější práci se systémem Greenstone. Praktické výsledky měly kromě demonstrace možností digitální knihovny v podobě netriviálních ukázkových sbírek zahrnovat také použitelné uživatelské rozhraní pro správu sbírek a lokalizaci rozhraní do českého jazyka.

Teoretická část této práce obsahuje stručný úvod do oblasti digitálních knihoven a přiblížení jejich rostoucího významu. V tomto kontextu je představen systém pro správu digitálních knihoven Greenstone. Jednotlivé kapitoly obsahují popis struktury systému a vysvětlují některé důležité principy použité při jeho návrhu a uplatňované při jeho fungování. Nabízejí tak obecný úvod založený na informacích obsažených v knihách a manuálech zabývajících se systémem a doplněný o některé vlastní poznatky získané při jeho používání.

Praktická část shrnuje výsledky práce vykonané při plnění úkolů zadání. Představuje nástroj Simple Collection Manager, sloužící jako uživatelské rozhraní k základní správě sbírek digitální knihovny a pro snadnou manipulaci s metadaty souborů těchto sbírek. Popis ukázkových sbírek demonstrujících možnosti digitální knihovny slouží k vysvětlení postupů použitých při jejich tvorbě. Některé problémy byly konzultovány s autory digitální knihovny Greenstone a prezentované výsledky mohou posloužit jako cenný zdroj při vytváření podobných sbírek a také pro hlubší pochopení principů fungování digitální knihovny. Při řešení této diplomové jsem se stal oficiálním správcem překladu českého rozhraní digitální knihovny Greenstone a dokončil jsem překlad pro verzi 2.50. Pro budoucí další přizpůsobení systému českým podmínkám byla zjištěna a popsána možnost zprovoznění stemmingu pro nové jazyky. Pro české uživatele byl

nad rámec této práce vytvořen jednoduchý uživatelský manuál popisující důležité součásti systému Greenstone a dostupné nástroje pro tvorbu sbírek. V závěru praktické části byly uvedeny popisy a způsoby použití dvou zásadních mechanismů využívaných digitální knihovnou Greenstone – pluginů a maker. Uvedené kapitoly obsahují shrnutí vlastních poznatků doplněné o informace z různých zdrojů a dovolují rychle pochopit a využívat klíčové prvky sloužící pro přizpůsobení chování digitální knihovny konkrétním potřebám.

Digitální knihovna Greenstone představuje kvalitní nástroj pro správu sbírek obsahujících desítky stejně jako miliony dokumentů. Během jejího používání jsem ocenil především kvalitu návrhu, která se projevuje ve velké obecnosti přístupu k řešeným problémům. Velká konfigurovatelnost ve všech ohledech umožňuje relativně snadné přizpůsobení aktuálním požadavkům při zachování fungování jádra, které má na starosti samotnou správu materiálů sbírek. Externí moduly napsané v jazyce Perl dovolují uživatelům měnit způsob zpracování materiálů, jednoduchý jazyk maker dovoluje provádět okamžitě změny obsahu vzhledu sbírek i celého uživatelského rozhraní. Greenstone je vytvářen s cílem použití v nejrůznějších zemích světa a tomu také odpovídá možnost překladů rozhraní a další přizpůsobení konkrétním potřebám. Všechny materiály jsou uchovávány v kódování UTF a dovolují tak vytvářet sbírky materiálů různých jazyků při současném zachování specifických znaků. Skutečnost, že systém je vyvíjen v akademickém prostředí na univerzitě Waikato na Novém Zélandu, přináší na jednu stranu nejistotu v podobě nezaručené garance výsledného produktu a poskytovaných služeb. Na druhou stranu otevřenost systému, spolupráce s celosvětovými organizacemi OSN a UNESCO a široká komunita uživatelů z celého světa zajišťují neustály vývoj a přizpůsobování aktuálním požadavkům. Z vlastní zkušenosti také mohu potvrdit velkou ochotu a snahu samotných autorů systému pomoci při řešení. Systém pro správu digitálních knihoven Greenstone je kvalitním nástrojem s dobrou perspektivou dalšího vývoje a širokého použití na celém světě.

Literatura

[1] Arms, William Y.: Digital Libraries, MIT Press, Cambridge 2000, ISBN:0-262-01880-8

[2] ARNO. URL: http://www.uba.uva.nl/arno (duben 2004)

[3] Bainbridge, D., McKay, D., Witten, Ian H.: Greenstone digital library developer´s guide. URL: http://prdownloads.sourceforge.net/greenstone/Develop-en.pdf (duben 2004).

[4] Bartošek, M.: Digitální knihovny, Datakon 2001. Dokument dostupný na URL http://www.ics.muni.cz/mba/dl-datakon01.pdf (duben 2004)

[5] Bartošek, M., Kovář, P.: Digitální knihovna fotografií MU. In: Mezinárodní konference RUFIS 2002, Sborník příspěvků, ISBN: 80-86510-40-9. VUT v Brně 2002, str. 2.

[6] Bőrner, K., Chen, Ch.: Visual Interfaces to Digital Libraries. Springer-Verlag, Berlin 2002.

[7] Bush, V.: As We May Think. Atlantic Monthly. July (1945), 101-108. Dokument dostupný na URL http://www.theatlantic.com/unbound/flashbks/computer/bushf.htm (duben 2004).

[8] California Digital Library: Archival Resource Key (ARK). URL: http://www.cdlib.org/inside/diglib/ark/ (duben 2004).

[9] CERN: CDSware. URL: http://cdsware.cern.ch/ (duben 2004).

[10] CNRI: Handle System. URL: http://www.handle.net/ (duben 2004).

[11] Digital Library Initiative, Phase 1 (DLI-1). URL: http://www.dli2.nsf.gov/dlione/ (duben 2004)

[12] Digital Library Initiative, Phase 2 (DLI-2). URL: http://www.dli2.nsf.gov/ (duben 2004).

[13] Don, K. J., Bainbridge, D., Witten, Ian H.: The design of Greenstone 3: An agent based dynamic digital library. URL: http://www.greenstone.org/manuals/gs3design.pdf (duben 2004).

[14] Dublin Core Metadata Initiative. URL: http://dublincore.org/ (duben 2004).

[15] EU-NSF Digital Library Working Group: Interoperability between Digital Libraries – Position Paper. URL: http://www.iei.pi.cnr.it/DELOS/REPORTS/interop.htm (duben 2004).

[16] Eclipse Foundation: Eclipse. URL: http://www.eclipse.org/ (duben 2004).

[17] Free Software Foundation: GDBM. URL: http://www.gnu.org/software/gdbm/gdbm.html (duben 2004).

[18] Free Software Foundation: Ispell. URL: http://www.gnu.org/software/ispell/ispell.html (duben 2004)

[19] Greenstone archives. URL: http://www.sadl.uleth.ca/nz/cgi-bin/library?a=p&p=about&c=gsarch (duben 2004).

[20] Greenstone Digital Library Software. URL: http://www.greenstone.org/cgi-bin/library (duben 2004).

[21] IETF: Uniform Resource Names (URN). URL: http://www.ietf.org/html.charters/urn-charter.html (duben 2004).

[22] Innovative Technology-Applied: i-Tor. URL: http://www.i-tor.org/en (duben 2004).

[23] Kosek, J.: XML pro každého: podrobný průvodce, Praha, Grada Publishing 2000.

[24] Library of Congress. URL: http://www.loc.gov (duben 2004).

[25] Library of Congress: American Memory historical collections. URL: http://memory.loc.gov/ (duben 2004).
[26] Library of Congress: National Digital Library Program (NDLP). URL: http://memory.loc.gov/ammem/dli2/html/lcndlp.html (duben 2004).
[27] Library of Congress – Z39.50 Maintenance Agency. URL: http://lcweb.loc.gov/z3950/agency/ (duben 2004).
[28] Library of Congress: MARC Standards. URL: http://www.loc.gov/marc/ (duben 2004).
[29] MIT, Hewlett-Packard: DSpace. URL: http://www.dspace.org/ (duben 2004).
[30] MyCoRe. URL: http://www.mycore.de/engl/index.html (duben 2004)
[31] New Zealand Digital Library. URL: http://nzdl.org/ (duben 2004)
[32] Object Management Group: Common Object Request Broker Architecture (CORBA). URL: http://www.omg.org/gettingstarted/corbafaq.htm (duben 2004).
[33] OCLC: PURL – Persistent URL. URL: http://purl.oclc.org/ (duben 2004).
[34] Open Archives Initiative. URL: http://www.openarchives.org/ (duben 2004).
[35] Open Archives Initiative: Open Archives Initiative Protocol for Metadata Harvesting. URL: http://www.openarchives.org/OAI/openarchivesprotocol.html (duben 2004)
[36] Open Society Institute: A Guide to Institutional Repository Software, 2nd edition. URL: http://www.soros.org/openaccess/software (duben 2004).
[37] Siever, E., Spainhour, S., Patwardhan, N.: Perl in a nutshell: a desktop quick reference. Beijing, O´Reilly, 1999.
[38] Sun Microsystems: Java Technology. URL: http://java.sun.com/ (duben 2004).
[39] University of Virginia, Cornell University: FEDORA. URL: http://www.fedora.info/ (duben 2004).
[40] Virius, Miroslav : Programování v C++. Praha, Nakladatelství ČVUT, 1998.
[41] W3C Consortium: HyperText Markup Language (HTML). URL: http://www.w3.org/MarkUp/ (duben 2004).
[42] W3C Consortium: Overview of SGML Resources. URL: http://www.w3c.org/MarkUp/SGML/ (duben 2004).
[43] W3C Consortium: XML. URL: http://www.w3c.org/XML/ (duben 2004).
[44] Witten, Ian H., Bainbridge, D.: How to build a digital library. Elsevier Science 2003, ISBN: 1-55860-790-0.
[45] Witten, Ian H., Bainbridge, D., Boddie, S., Don, K. J., McPherson, J. R.: Inside Greenstone collections. URL: http://www.greenstone.org/docs/inside_greenstone.pdf (duben 2004)
[46] Witten, Ian H., Bainbridge, D., Panter, G., Boddie, S.: The Greenstone plugin architecture. URL: http://www.cs.waikato.ac.nz/~ihw/papers/02-IHW-etal-Thegreesoneplugin.pdf (duben 2004).
[47] Witten, Ian H., Boddie, S.: Greenstone digital libray installer´s guide. URL: http://prdownloads.sourceforge.net/greenstone/Install-en.pdf (duben 2004).
[48] Witten, Ian H., Boddie, S., Thompson, J.: Greenstone digital library user´s guide. URL: http://prdownloads.sourceforge.net/greenstone/User-en.pdf (duben 2004).
[49] Witten, Ian H., McNab, Rodger J., Boddie, S., Bainbridge, D.: Greenstone: A Comprehensive Open-Source Digital Library Software System, URL: http://portal.acm.org/citation.cfm?id=336650&coll=Portal&dl=ACM&CFID=20669712&CFTOKEN=15459524 (duben 2004).

[50] Witten, Ian H., Moffat, A., Bell, T.: Managing Gigabytes: compressing and indexing documents and images. Morgan Kaufmann 1999, second edition.

[51] Zhang, A.: Customizing the Greenstone User Interface. Washington Research Library Consortium, August 2003. URL: http://www.wrlc.org/dcpc/UserInterface/interface.htm (duben 2004).

[52] Zvon: RFC1321 (MD-5). URL: http://www.zvon.org/tmRFC/RFC1321/Output/ (duben 2004).

Přílohy

Příloha I – Transformovaná metadata pro sbírku DKF

```
<FileSet>
      <FileName>00200009.000.dkf</FileName>
      <Description>
            <Metadataname="dc.Identifier">00200009.000</Metadata>
            <Metadata name="dc.Gallery">uvt</Metadata>
            <Metadata name="dc.Title">Instalace EC-1033</Metadata>
            <Metadata name="dc.Description">Instalace prvního velkého
            počítače univerzity, počítače EC-1033, v prostorách
            Laboratoře počítacích strojů VUT Brno na Třídě Obránců
            míru (dnešní Údolní ulice).</Metadata>
            <Metadata name="dc.Order">002</Metadata>
            <Metadata name="dc.Previous">00200014.000</Metadata>
            <Metadata name="dc.Next">00200005.000</Metadata>
      </Description>
</FileSet>
<FileSet>
      <FileName>00200009.001.1.jpg</FileName>
      <Description>
            <Metadata name="dc.GroupIdentifier">00200009.000
                  </Metadata>
            <Metadata name="dc.Identifier">00200009.001</Metadata>
            <Metadata name="dc.Title">Instalace EC-1033</Metadata>
            <Metadata name="dc.Order">9</Metadata>
            <Metadata name="dc.Description">Instalace snímače
                                                štítků.</Metadata>
            <Metadata name="dc.Next">00200009.002</Metadata>
            <Metadata name="dc.LargeFileName">00200009.001.1.jpg
                  </Metadata>
            <Metadata name="dc.LargeWidth">1024</Metadata>
            <Metadata name="dc.LargeHeight">714</Metadata>
            <Metadata namme="dc.NormalFileName">00200009.001.2.jpg
                  </Metadata>
            <Metadata name="dc.NormalWidth">640</Metadata>
            <Metadata name="dc.NormalHeight">446</Metadata>
            <Metadata name="dc.ThumbFileName">00200009.001.3.jpg
                  </Metadata>
            <Metadata name="dc.ThumbWidth">160</Metadata>
            <Metadata name="dc.ThumbHeight">112</Metadata>
      </Description>
</FileSet>
```

Příloha II – Popis nástroje Librarian

1 Librarian – nástroj pro správu kolekcí

Tvůrci sbírek digitální knihovny Greenstone měli dlouhou dobu k dispozici přístup k funkcím nabízeným digitální knihovnou pouze pomocí manuálního spouštění Perlovských skriptů a ruční editace konfiguračních souborů. Hlavně začínajícím uživatelům bez potřebných znalostí však díky tomu zůstávala značná část možností systému skryta. Tvůrci systému od roku 1999 pracovali na kvalitním grafickém uživatelském rozhraní, které by umožnilo snadné vytváření a správu sbírek a zároveň zpřístupnilo většinu funkcí snadno uchopitelným způsobem. Spolu s vydáním Greenstone verze 2.40 (v roce 2003) byla dána k dispozici také první verze tohoto rozhraní, nazvaného *Librarian*. Ačkoliv tato verze obsahovala řadu chyb, jednalo se o velké zlepšení oproti předchozímu stavu. Librarian umožňuje kompletní správu sbírek – od jejich založení přes přiřazování metadat až po samotný import dokumentů a vytváření sbírek. Jedná se tedy o aplikaci, která nahrazuje dřívější úpravu nastavení na různých místech jednotným a uživatelsky příjemnějším rozhraním. Nástroj je stále ve vývoji, s každou novou verzí systému Greenstone se objevuje i nová verze aplikace Librarian. Následující kapitoly popisují Librarian, který byl součástí instalace Greenstone verze 2.41.

Librarian je napsán v programovacím jazyce Java a ke svému spuštění potřebuje, funkční instalaci Java runtime environment (instalaci prostředí je možné stáhnout z oficiálních stránek http://java.sun.com/j2se/downloads.html). Aplikace Librarian se nachází v podadresáři GLI adresáře s instalací Greenstone. V adresáři GLI pak lze nalézt skripty (dávkové soubory) sloužící pro spouštění a správu nástroje Librarian:

- `makegli.bat` – zkompilování Librarianu
- `gli.bat` – spuštění Librarianu
- `document.bat` – vytvoření dokumentace k nástroji Librarian
- `clean.bat` – vyčištění adresáře GLI

Dávka `gli.bat` zkontroluje, zda jsou nainstalovány potřebné nástroje (Java runtime environment, Perl, Greenstone) a zda je Librarian správně zkompilován. Pokud všechno proběhne v pořádku, je aplikace Librarian spuštěna.

1.1 Librarian – první pohled

Po spuštění aplikace se objeví hlavní okno. Kromě hlavního menu v záhlaví okna jsou všechny ovládací prvky rozmístěny do čtyř záložek – *Gather*, *Enrich*, *Design* a *Create*. Každá z těchto záložek pokrývá jednu z oblastí souvisejících se správou kolekcí:

- *Gather* – shromaždování a organizace souborů určených pro import do digitální knihovny.
- *Enrich* – přiřazování a úprava metadat.
- *Design* – nastavování parametrů ovlivňujících jak proces budování kolekce, tak i následnou prezentaci dokumentů. Obsahuje nástroje pro nastavování klasifikátorů, výběr vytvářených indexů, návaznosti na jiné kolekce a mnoho dalších.
- *Create* – spouštění budování kolekce a zobrazení výstupů z jednotlivých fází procesu.

Pokud není otevřena žádná kolekce, jsou funkční pouze některé položky hlavního menu a první záložka (Gather). Máme-li již nějakou kolekci otevřenou, můžeme ji, kromě úprav popsaných v následujících kapitolách, například exportovat na CD-ROM nebo smazat.

1.2 Založení kolekce

Novou kolekci vytvoříme přes hlavní menu (`File->New`). Objeví se dialogové okno, které vyžaduje zadání základních údajů o vytvářené kolekci – názvu (*collection title*) a popisu obsahu kolekce (*description of content*). Máme také možnost zvolit, zda nová kolekce bude vytvořena na základě nějaké již existující (*Base this collection on*). V případě, že vybereme tuto možnost, bude do námi vytvářené kolekce zkopírována většina nastavení té kolekce, kterou používáme jako vzor. Jedná se především o počet a druh indexů, nastavení klasifikátorů, formát začleňovaných dokumentů a výchozí způsob prezentace. Tato možnost je zvláště výhodná v případě, že vytváříme řadu kolekcí, které jsou stejně nebo podobně organizovány a prezentovány – použitím již vytvořených nastavení můžeme ušetřit mnoho času.

Zadané údaje by měly být v anglickém jazyce, protože ten je výchozí u všech kolekcí. Angličtina ale není omezením, které by zabraňovalo použití jiných jazyků. Slouží pouze jako styčný bod u cizojazyčných sbírek, aby i uživatel, který nezná jazyk sbírky, byl schopen zjistit, čeho se sbírka týká. K zadaným údajům máme možnost později na záložce Design doplnit překlady do libovolných jazyků a přizpůsobit tak prezentaci kolekce jazyku většiny budoucích uživatelů. Na tomto dialogovém okně se mohou také objevit další pole, požadující například zadání e-mailu autora, zkratky názvu kolekce a podobně. Aby uživatel nemusel tyto údaje zadávat stale znovu, je možné je zadat v nastaveních Librarianu (`File->Preferences`). Všechny informace lze později editovat na záložce Design.

Po zadání a potvrzení základních údajů se objeví druhé dialogové okno, nabízející metadatové sady (*metadata sets*), které budou přidány ke kolekci a budou určovat metadatové schéma, podle kterého budeme u souborů kolekce zadávat metadata. K dispozici jsou dvě základní schémata – ukázková sada pro vývoj knihoven (*developement library subset metadata example*) a *Dublin Core 1.1*. Další sady lze přidávat – vytvořením souboru s popisem metadatového schématu a umístěním do patřičného adresáře (více v kapitole o přiřazování metadat). Nemusíme však vybrat žádné metadatové schéma a učinit tak případně až později při konfiguraci sbírky.

1.3 Shromáždění materiálů – záložka Gather

Pro shromáždění materiálů (souborů různých formátů), které hodláme zařadit do kolekce, můžeme použít nástroj *Gatherer*. Příslušná záložka (*Gather*) je rozdělena na dvě části – vlevo máme možnost procházet soubory na disku, vpravo je zobrazen obsah adresáře, sloužícího pro import do naší kolekce. Gatherer umožňuje vytvářet adresáře a do nich přetahováním položek z levého podokna kopírovat soubory nebo celé adresáře. V případě, že tento způsob není vyhovující (testovaná verze Librarianu měla občas problémy s kopírováním položek), můžeme stejného výsledku dosáhnout prostým nakopírováním souborů do adresáře `import` dané kolekce (nachází se v podadresářích `collect/[nazev_kolekce]/import` adresáře s instalací digitální knihovny Greenstone).

1.4 Přiřazení metadat – záložka Enrich

Funkce zpřístupněné záložkou *Enrich* umožňují snadno editovat metadata u dokumentů určených k zařazení do kolekce. Metadata jsou ukládána do souborů s názvem `metadata.xml`, které jsou umístěny na jednotlivých úrovních adresářové struktury určené pro import. Pokud nepoužíváme nástroj Librarian, musíme soubory `metadata.xml` editovat s použitím jiných nástrojů.

Jak již bylo zmíněno dříve, asociování metadat (MD) s dokumenty úzce souvisí s vybranými metadatovými schématy (MDS). Základní schéma, které se u kolekcí používá, je interní MDS Greenstone založené na MD, které lze extrahovat ze souborů samotných (jméno, typ, velikost). Mezi ně se řadí také MD přiřazená pluginy bez určení schématu (taková MD jsou pak opatřena prefixem 'ex' – například `ex.Title`).

Kromě tohoto základního schématu můžeme zvolit libovolný počet jiných MDS, které určí typ a formát MD asociovaných se soubory. Jednotlivá MDS můžeme také upravovat (přidávat/rušit elementy, atributy), vytvářet, exportovat, či přiřazovat jak celým kolekcím, tak i jednotlivým adresářům v rámci kolekce. Všechny operace s MDS jsou zahrnuty v hlavním menu pod položkou *Metadata Sets*. Pokud máme vybrána MDS, která chceme používat, můžeme začít s přiřazováním metadat. Ta jsou klíčová pro další kroky v editaci kolekce a přímo ovlivňují další možnosti – například vytváření indexů a tím i kvalitu vyhledávání v budoucí kolekci.

Záložka *Enrich*, určená pro manipulaci s metadaty, je rozdělena na tři oddíly. V levé části okna se nachází seznam souborů kolekce, který ukazuje umístění souborů v adresářové struktuře. V pravé horní části okna je tabulka zobrazující metadata přiřazená danému souboru/složce. Oblast vpravo dole pak slouží k editaci hodnot – umožňuje jednak zadávat nové a nebo vybírat z již zadaných hodnot pro daný element. MD lze přiřazovat jak jednotlivým souborům, tak i celým složkám. Přiřazení metadat složce způsobí kaskádové přidělení zadaných MD všem souborům, ležícím v hierarchii pod danou složkou. Takto lze namísto pracné editace MD u každého souboru zadat společné hodnoty elementů najednou. MD mohou být jednak přiřazována akumulovaně – tedy pro každý soubor může být zadáno více hodnot u stejného elementu a nebo exkluzivně – daný element pak může mít nejvýše jednu hodnotu. Vhodnou kombinací akumulovaných a exkluzivních metadat lze například detailněji specifikovat u souborů MD přiřazená na úrovni složek, nebo je nahrazovat jinými hodnotami. Jak již bylo zmíněno, zvolené MDS a zadání hodnot jednotlivých elementů ovlivňuje další možnosti práce se sbírkami.

1.5 Správa kolekce – záložka Design

Záložka Design obsahuje všechna důležitá nastavení, která ovlivňují jak interní strukturu kolekce (množství, typy a úrovně indexů, způsoby importování dokumentů, nabízené možnosti vyhledávání ...), tak i její prezentaci. Většina nastavení zpřístupněných na této záložce je uložena v hlavním konfiguračním souboru sbírky (`collect.cfg` umístěný v adresáři `etc/`). Stejné úpravy můžeme provádět také manuálně, výhodou použití Librarianu je však rychlý přístup k veškerým důležitým nastavením a uživatelsky příjemnější editace včetně nápověd k jednotlivým možnostem. Záložka je rozdělena na tři základní oblasti, přičemž poslední oblast se v některých případech ještě dále dělí:

- **levý panel** obsahuje seznam nastavení zpřístupněných na záložce Design, kliknutím na položku se aktivuje editace parametrů
- **pravý horní panel** obsahuje krátký popis a nápovědu k právě editovanému nastavení
- **pravý dolní panel** slouží k samotné editaci parametrů

Pokud vytvoříme novou kolekci, jsou všechna nastavení „standardní“ – umožňují tedy sice vytvořit kolekci aniž bychom museli cokoliv editovat, na druhou stranu neřeší neobvyklé požadavky jako například zpracování neobvyklých formátů souborů, odlišná metadatová schémata a indexy na nich založené nebo i vzhled kolekce a jednotlivých dokumentů v ní obsažených. Nezkušený uživatel tedy může vše nechat tak jak je a zkoušet vytvářet jednoduché sbírky, případně zkoušet upravovat jen některá nastavení. Čím větší nároky na sbírku jsou kladeny, tím větších zásahů bude potřeba.

Pokud při zakládání kolekce zvolíme možnost vytvoření na základě již existující kolekce (viz kapitola Založení kolekce), budou standardní nastavení přepsána nastaveními kolekce použité jako vzor. Tato možnost je obzvláště výhodná pokud vytváříme stejně utvářené sbírky s množstvím vlastních nastavení. Způsob editace a význam jednotlivých nastavení je detailně popsán v následujících kapitolách.

1.5.1 Nastavení obecných údajů o kolekci

Obecné údaje o kolekci jsou zahrnuty pod položkou *General*. Zde máme možnost nastavit kontakt na tvůrce a správce kolekce, základní omezení přístupu ke kolekci (je-li veřejně přístupná či nikoliv), název a popis. Jedná se o údaje, které bylo možno zadat při vytváření kolekce, na tomto místě je můžeme editovat či doplňovat. Navíc je zde možnost přiřadit kolekci „ikony“ – obrázky, které budou reprezentovat kolekci v rámci knihovny:

- **hlavička kolekce** (*URL to about page icon*) – tento obrázek se zobrazuje v hlavičce kolekce – ve standardním nastavení vlevo nahoře nad navigačním panelem. Kliknutím na tento obrázek se otevírá stránka s obecnými údaji o kolekci. Pokud není žádný obrázek přidělen, při prohlížení kolekce se místo něj zobrazí název kolekce (obdobně to platí i u zástupce kolekce).
- **zástupce kolekce** (*URL to home page icon*) – ikona, která zastupuje kolekci při výpisu kolekcí spravovaných danou instancí digitální knihovny Greenstone.

1.5.2 Výběr a nastavení vhodných pluginů

Položka *Document Plugins* (v levém výběrovém panelu) zastupuje soubor nástrojů pro prácí s pluginy – přidávání/odebírání a konfiguraci. Pluginy jsou jedním ze základních kamenů, ovlivňujících podobu a strukturu kolekce. Jedná se o moduly napsané v jazyce Perl, které umožňují různým způsobem zpracovávat soubory určené k začlenění do sbírky, přiřazovat k nim metadata a řídit budování kolekce. Popisem, tvorbou a podrobnou konfigurací pluginů se zabývá kapitola Pluginy a jak je vytvářet, v této kapitole budou uvedeny pouze obecné informace potřebné k jejich správnému používání. Okno pro nastavení pluginů obsahuje:

- seznam pluginů pro danou kolekci
- tlačítka pro změnu pořadí pluginů v seznamu
- tlačítka na přidání/konfiguraci/odebírání pluginů

Seznam obsahuje výčet všech pluginů, které budou použity při vytváření kolekce. Na každém řádku je uveden jeden plugin včetně nastavených parametrů. Seznam je dále rozdělen horizontální čarou. Oblast nad čarou plně podléhá změnám uživatele – může libovolné položky přidávat či odebírat a nastavovat parametry. Oblast pod čarou je vyhrazena pluginům, které jsou nezbytně nutné pro vytvoření kolekce. Konkrétně se jedná o plugin *ArcPlug*, který se využívá při budování kolekce (prochází strukturou naimportovaných souborů a předává je dalšímu zpracování) a plugin *RecPlug*, který slouží pro zpracování metadat při importu souborů. Pluginy v oblasti pod čarou není možné odebírat, lze pouze měnit jejich parametry.

V rámci seznamu lze také pomocí čtyř tlačítek umístěných po jeho pravé straně měnit pořadí položek. Pořadí pluginů může často ovlivnit výstupy, které vytváří – čím výše je plugin na seznamu, tím dříve mu bude umožněno zpracovat soubor na vstupu. Ovládací panel umístěný vpravo dole nám umožňuje přidávat nové pluginy a konfigurovat či odebírat stávající.

V následujících podkapitolách bude stručně popsáno co jsou pluginy a jak je základně konfigurovat. Dále uvedené informace by měly umožnit takový náhled na správu pluginů, aby bylo možné využít možností, které poskytují.

1.5.2.1 Co je to plugin

Jak již bylo zmíněno dříve, plugin je modul (soubor s blokem kódu napsaným v jazyce Perl), který ovlivňuje způsob, jakým systém Greenstone pracuje s dokumenty při importu a budování kolekce. Pluginy umožňují flexibilně reagovat jak na potřeby importu různých formátů souborů, tak i na požadavky na jejich zpracování. Bylo by sice možné každý soubor vložit do kolekce „tak jak je“ a používat knihovnu jen jako jistý druh archivu, ale tím bychom se ochudili o možnosti nabízené digitálními knihovnami – například o snadné vyhledávání a třídění, které jsou založené na klasifikaci a indexování obsahu dokumentů.

Součástí instalace systému Greenstone je velké množství pluginů, přičemž další můžeme libovolně vytvářet a přidávat. Opomineme-li speciální pluginy sloužící pro zpracování interních metadat nebo řídící budování kolekce (jako jsou například ArcPlug, RecPlug nebo GAPlug), je většina určena pro zpracování a import různých formátů souborů. U dokumentů obsahujících text většinou provádějí konverzi obsahu do textové podoby (například z formátu `PDF` do čistého textu) za účelem vytvoření indexů a umožnění vyhledávání. Dále z této textové verze vytváří dokument, který lze v rámci kolekce prohlížet, protože však často nejsou schopné zvládnout formátování původního dokumentu, přidávají do kolekce i soubor v původním formátu.

Některé pluginy také umí extrahovat metadata obsažená v souboru a přidávat je k metadatům přiřazeným externě (například *HTMLPlug* umí získat jméno autora nebo rozdělit dokument na kapitoly podle metadat obsažených v HTML souboru), asociovat další soubory (u pluginu HTMLPlug jsou to obrázky), formátovat vzhled dokumentu v rámci sbírky a další.

Při importu/budování kolekce se procházejí jednotlivé zpracovávané soubory a předkládají se postupně jednotlivým pluginům. Pokud je plugin schopen soubor zpracovat, učiní tak, v opačném případě jej odmítne a soubor je nabídnut následujícímu pluginu. Pluginy tedy slouží jako moduly rozšiřující základní možnosti systému a umožňující zpracovávat jakýkoliv formát souboru takovým způsobem, jaký požadujeme.

1.5.2.2 Význam pořadí pluginů

Při importu souborů jsou jednotlivé soubory postupně předkládány pluginům uvedeným v seznamu. Pořadí rozhoduje o tom, který plugin se k souboru „dříve dostane“. Zdálo by se, že pokud každý plugin zpracovává jen určitý formát souborů, nemělo by ke konfliktu dojít (nebudeme pravděpodobně používat dva pluginy zpracovávající stejný formát). Konflikt i přesto může nastat.

Jedním z příkladů může být kolekce, ve které chceme mít jednak obrázky a jednak dokumenty obrázky obsahující. Dokumenty s obrázky budou vytvářeny na základě importu HTML stránek, obrázky pak ze souborů neodkazovaných z žádného HTML dokumentu. Máme k dispozici plugin na zpracování HTML souborů (HTMLPlug je součástí instalace a umožňuje asociování obrázků, zároveň však blokuje zpracování obrázků dalšími pluginy) a námi vytvořený *ImageImportPlug*, který importuje obrázky samotné (aby zjistil, které obrázky má importovat, prohlíží metadata asociovaná se souborem). Kdybychom v této modelové situaci umístili HTMLPlug před ImageImportPlug, výsledkem by byla kolekce obsahující pouze dokumenty s vloženými obrázky, ale nikoliv obrázky jako samostatné dokumenty. Pokud ale pořadí pluginů změníme, umožníme importovat jak obrázky, tak i dokumenty obrázky obsahující.

Toto je pouze jeden z jednoduchých příkladů, ve skutečnosti se mohou vyskytnout závažnější konflikty, které je těžké vyřešit. Často však příčinou „nevysvětlitelného“ chování importu dokumentů bývá právě jen chybné pořadí pluginů.

1.5.2.3 Konfigurace pluginů

Chování jednotlivých pluginů do značné míry ovlivňuje nastavení jejich parametrů. Jelikož pluginy vytvářejí podobně jako u dědičnosti hierarchii s pluginem *BasPlug* jako společným předkem, jsou některé parametry pro všechny pluginy stejné. Jinak také mohou obsahovat specifické parametry.

Librarian umožňuje oproti klasickému ručnímu editování (textovému seznamu přepínačů aktivujících jednotlivá nastavení) uživatelsky daleko příjemnější a přehlednější ovládání. Po otevření okna konfigurace pluginu se zobrazí výčet jednotlivých parametrů s přepínači, určujícími, zda má být daný parametr použit či nikoliv. U některých parametrů je také možné zadat další argumenty, které budou ovlivňovat fungování pluginu. Navíc jsou rozlišeny „základní“ a „přidané“ parametry (tedy ty zděděné od předků a ty vlastní danému pluginu). K jednotlivým parametrům je také poskytnuta nápověda, která se objeví, setrvá-li nad názvem parametru kurzor myši. Příklady parametrů, které je možné nastavit:

- **vstupní kódování** – Greenstone používá z důvodů nezávislosti na znakových sadách pro interní reprezentaci znaků kódování UTF-8. Většina pluginů se snaží automaticky určit kódování zpracovávaného dokumentu, v některých případech je však nutné kódování explicitně zadat pomocí parametru.
- **jazyk dokumentu** – v jakém jazyce je daný dokument napsán.
- **extrahovat ...** – různé parametry určují, zda se má plugin pokusit z textu extrahovat metadata (například e-mailové adresy, letopočty, zkratky ...).
- **obrázek dokumentu** – určení obrázku, který bude sloužit jako „přebal“ dokumentu (zobrazí se vedle hlavičky dokumentu při prohlížení kolekce).

1.5.3 Typy hledání

Pod položkou Typy hledání (*Search types*) se ukrývá pokročilé nastavení, které určuje jakým způsobem mají být vytvářeny indexy pro vyhledávání ve sbírce. Greenstone umožňuje využívat dva různé systémy na kompresi textů a vytváření indexů – MG a MGPP.

MG (zkratka pro *Managing gigabytes*) je fulltextový indexační systém, který vytváří indexy na úrovni dokumentů. Umožňuje zadávat dotazy ve formě regulárních výrazů a efektivně vyhledávat v speciálních komprimovaných indexech. MGPP je pak rozšířením systému MG, které poskytuje indexování na úrovni slov, zpřístupňuje prohledávání na základě kontextu (*proximity searching*) a prohledávání polí.

Greenstone standardně využívá pro vytváření indexů systém MG. Změna nastavení u položky Typy hledání (zaškrtnutím možnosti *Enable Advanced Searches*) nám umožňuje zvolit jako indexační systém MGPP a vybrat, jaké druhy hledání mají být k dispozici:

- **plain** – vyhledávání pomocí regulárních výrazů (podobné jako u standardně použitého systému MG).
- **form** – prohledávání polí s využitím booleovských výrazů a podmínek vymezujících vlastnosti a kontext hledaného výrazu.

Pokud zvolíme oba typy hledání, rozhoduje jejich pořadí o tom, který z nich bude u kolekce použit jako základní. Mezi možnými typy hledání lze také přepínat za běhu digitální knihovny – pomocí nastavení u dané sbírky.

1.5.4 Definování indexů

Abychom měli možnost využívat výhody rychlého a kvalitního vyhledávání v rámci sbírky, musíme určit, jaká metadata (atributy) má systém použít k vytvoření indexů. Možnosti nastavení a vzhled okna pro editaci parametrů indexů se různí podle typu hledání, který jsme vybrali (viz předchozí podkapitola).

Používáme-li základní typ hledání (indexy vytvořené systémem MG), zobrazí se nám v okně seznam indexů k vytvoření a dále ovládací prvky k přidání/odebrání indexů. U každého indexu je třeba určit tři parametry: jméno (*Index name*), co má být indexováno (*Build index on*) a úroveň (*At the level*).

Zatímco **jméno** je určeno pro uživatele (objevuje se například při výběru indexu pro vyhledávání v rámci kolekce), ovlivňují předmět a úroveň indexace vlastnosti samotného hledání. **Předmětem indexace** mohou být jednak metadata (autor, název, zdroj ...), tak i samotné fyzické atributy (text). Navíc je možné vytvořit index s využitím více atributů. **Úroveň** může nabývat tří hodnot: *document*, *paragraph*, *section*. Každá z nich poskytuje jinou úroveň náhledu – od dokumentu jako celku až po jednotlivé jeho odstavce. V případě podrobnějšího členění dokumentu však musí být jeho vnitřní struktura explicitně určena. Volbou různé úrovně indexů například můžeme umožnit vyhledávání buď v názvech dokumentů či jednotlivých jejich kapitol.

Pokud v Typech vyhledávání zvolíme rozšířené vyhledávání (a tím použití indexačního systému MGPP), bude okno dále rozděleno na dvě záložky. Na první (*Manage indexes*) se kromě přehledu indexů nachází ovládací prvky pro přidávání/modifikaci/odebírání indexů, na druhé (*Manage levels*) pak ovládací prvky pro volbu úrovně indexů – úrovně se zde aplikují na všechny indexy. Rozdílem oproti dříve popsanému zadávání indexů je možnost zadat k

indexování pouze jeden atribut/pole (s výjimkou možnosti *allfields* která umožňuje prohledávat všechna indexovaná pole).

1.5.5 Částečné indexy

Nastavení pod položkou *Partition indexes* umožňují definovat omezení při prohledávání indexů a tím vlastně vytvářet indexy nové. Okno obsahuje tři záložky: definování filtrů (*Define filters*), přiřazení rozdělení (*Assign partitions*) a přiřazení jazyků (*Assign languages*).

Definování filtrů slouží k specifikaci omezení prohledávaných dokumentů pomocí omezení kladených na hodnoty atributů. Na záložce je kromě seznamu již nadefinovaných filtrů také sada ovládacích prvků sloužících pro vytváření filtrů nových. Definice filtru obsahuje název filtru, určení atributu dokumentu, na který jsou kladena omezení, a regulárního výrazu omezení popisujícího. Můžeme například specifikovat filtr, který bude dokumenty omezovat podle určitého autora, roku vytvoření nebo podle formátu.

Záložka přiřazení rozdělení dovoluje na základě vytvořených filtrů (jednoho nebo více) vytvářet výseky z indexů, které slouží k přesnějšímu omezení při vyhledávání. Na záložce máme možnost vybrat libovolný počet dříve definovaných filtrů a vytvořit z nich rozdělení a pojmenovat je. Záložka přiřazení jazyků funguje obdobně, ovšem jako filtry slouží určení požadovaného jazyka dokumentu.

Uplatnění částečných indexů spočívá v řízeném omezení při vyhledávání v rámci sbírky. V závislosti na počtu vytvořených rozdělení se na stránce určené pro hledání v rámci kolekce zobrazí kromě nabídky indexu samotného také nabídky rozdělení indexů a omezení jazyka dokumentů. Celkový počet „částečných indexů“, které můžeme získat různými aplikacemi rozdělení, je pak roven počtu kombinací „normálních“ indexů, rozdělení a omezení jazyka dokumentu. Například pro kolekci se třemi indexy, dvěma definovanými rozděleními a dvěma omezeními podle jazyka dostaneme celkem 12 možných různých částečných indexů.

1.5.6 Vyhledávání napříč kolekcemi

Pod položkou *Cross-Collection Search* nalezneme možnost, jak do indexu vytvářeného pro právě editovanou kolekci přidat data z indexů jiných kolekcí. Důležitým předpokladem je, aby všechny zahrnuté kolekce měly stejnou strukturu indexů – tj. aby přinejmenším indexovaly stejné atributy. Jakékoliv odchylky, například v podobě rozdělení indexů (částečných indexů) nebo nesrovnalostí v indexovaných atributech, mohou vést k neúplnému či dokonce nesprávnému fungování vyhledávání.

Vyhledávání napříč kolekcemi nám zpřístupňuje materiály z různých kolekcí a dovoluje například vytvářet speciální „vyhledávací kolekce“. Ty mohou sloužit jako centra shromažďující indexy a umožňující vyhledávání v relevantních sbírkách. Výsledkem hledání v takovém indexu pak je seznam odkazů, vedoucí k dokumentům různých sbírek – kliknutím na odkaz se dostáváme nejen k dokumentu samotnému, ale také do sbírky, která jej obsahuje.

Okno pro přidávání kolekcí do vyhledávání obsahuje jednoduchý seznam aktivních kolekcí s ovládacím prvkem určujícím, zda daná kolekce má či nemá být zahrnuta do vyhledávání. Vždy je vybrána právě editovaná kolekce, přičemž je možné přidat libovolný počet dalších již existujících kolekcí.

1.5.7 Nastavení klasifikátorů

Vyhledávání je pouze jedním ze způsobů získávání materiálů v rámci digitálních knihoven. Často má uživatel místo hledání konkrétního dokumentu zájem procházet kolekcí a zajímavé materiály objevovat. Greenstone za tímto účelem používá klasifikátory (*classifiers*) – indexy sloužící pro procházení dokumenty. Podobně jako u indexů pro vyhledávání jsou klasifikátory vytvářeny na základě metadat příslušejících jednotlivým dokumentům. Součástí instalace je řada standardních klasifikátorů poskytujících různé možnosti organizace dokumentů – od jednoduchého seznamu až po složité stromové hierarchie. Podobně jako pluginy lze i klasifikátory definovat vlastní a zpřístupňovat je pro použití v kolekcích. Jedná se však o vysoce specializovanou činnost, kterou se tento dokument nebude zabývat.

Librarian umožňuje přidávat, konfigurovat a odebírat klasifikátory pomocí ovládacích prvků umístěných pod položkou *Browsing Classifiers*. V horní části okna je zobrazen seznam použitých klasifikátorů včetně nastavení, v dolní části pak možnost výběru konkrétního klasifikátoru a tlačítka pro jeho přidání, odebrání a konfiguraci. Konkrétní nastavení jednotlivých klasifikátorů se značně liší, u většiny však můžeme nalézt položku *metadata*, která určuje nad kterými metadaty má být vybudován index a *buttonname* určující název tlačítka, pod kterým bude klasifikátor dostupný v navigačním panelu kolekce. Další nastavení pak mohou ovlivňovat vzhled, úroveň podrobnosti nebo návaznost na jiné klasifikátory. Funkční klasifikátory jsou přístupné přes odkazy umístěné na navigačním panelu kolekce.V následujícím seznamu jsou shrnuty klasifikátory, které jsou součástí instalace:

- *AZCompactList* – varianta *AZList*u, která zobrazuje seznam dokumentů abecedně setříděných podle hodnot zadaného atributu. Více dokumentů se stejnou nebo podobnou hodnotou atributu je sloučeno do skupiny pro větší přehlednost.
- *AZCompactSectionList* – podobné jako *AZCompactList*, indexuje ale sekce dokumentu (kapitoly, odstavce) namísto názvů.
- *AZList* – seznam dokumentů abecedně setříděných podle hodnoty zadaného atributu. Dokumenty jsou dále rozděleny do skupin podle rozdělení abecedy.
- *AZSectionList* – seznam sekcí dokumentů abecedně setříděných podle hodnoty zadaného atributu.
- *DateList* – v závislosti na datu vytvoření jsou dokumenty rozděleny do skupin. Jednoduchá dvouúrovňová hierarchie umožňující snazší orientaci.
- *Hierarchy* – Hierarchický klasifikátor. Na základě struktury zadané v textovém souboru rozděluje dokumenty do skupin a vytváří tak (teoreticky) neomezený strom.
- *HTML* – „slepý“ klasifikátor. Slouží pouze jako odkaz na HTML stránku.
- *List* – nejjednodušší klasifikátor – seznam dokumentů.
- *Phind* – umožňuje vyhledávání frází v dokumentech kolekce.
- *SectionList* – seznam sekcí dokumentů.

Každé kolekci je možné přiřadit libovolný počet klasifikátorů. Zadáním různých kritérií vytváření indexů (tj. metadat, podle kterých budou dokumenty odkazovány) dovoluje prezentovat dokumenty kolekce z více pohledů – například pomocí sdružování do hierarchií podle různých společných vlastností.

1.5.8 Úprava zobrazení dokumentů a seznamů dokumentů

Jednou z velkých předností digitální knihovny Greenstone je relativně snadné formátování výstupu – tedy vzhledu kolekcí, klasifikátorů a dokumentů. Podoba stránek knihovny je totiž generována dynamicky („*on the fly*“) za použití maker a „proměnných“. Tento přístup jednak umožňuje, aby vzhled kolekcí odpovídal přání uživatele a současně se v závislosti na prostředí (jazyk, různé kolekce) lišil. Úpravy vzhledu lze rozdělit do dvou kategorií: kolekce jako celek a obsah kolekce.

Librarian poskytuje nástroje na editaci vzhledu obsahu. Pod položkou *Formating output* nalezneme seznam přiřazených formátovacích řetězců, panel pro výběr předmětu formátování (text dokumentu, klasifikátory ...), panel pro detailnější specifikaci předmětu formátování (například horizontálně vypisovaný seznam) a pole pro zadávání formátovacích řetězců. Všechny formátovací řetězce jsou ukládány do konfiguračního souboru kolekce (`etc/collect.cfg`) a Librarian stejně jako u většiny ostatních nastavení slouží jen jako nástroj k jejich pohodlnější editaci.

Jelikož kolekce a dokumenty jsou prezentovány jako HTML stránky, patří mezi základní formátovací prvky standardní HTML tagy. Příslušná data (texty dokumentů, hodnoty metadat) jsou pak reprezentována „proměnnými“ – speciálními řetězci, které jsou při běhu digitální knihovny dynamicky nahrazovány konkrétními hodnotami. Dodatečné informace závislé na nastavení prostředí – různojazyčné popisky, aktuální odkazy na zdroje v rámci knihovny apod. – jsou zastoupeny pomocí maker.

1.5.8.1 Proměnné při formátování

Proměnné zastupují na dokumentu závislé atributy, případně formátování z těchto atributů odvozené. Proměnná je odkazována identifikátorem v hranatých závorkách – pokud například používáme pro popis dokumentů MDS Dublin Core, pak výraz `[dc.Title]` bude při generování stránky nahrazován konkrétní hodnotou atributu `dc.Title` daného dokumentu. Librarian nabízí vložení některých proměnných, které se u dokumentů běžně vyskytují (odkaz na dokument, vložení obsahu dokumentu). V následujícím výčtu jsou shrnuty základní proměnné, které lze při psaní formátovacích řetězců využít:

- **[název_metadat]** – při vytváření stránek je tato proměnná nahrazena hodnotou daného elementu (například `[dc.Title]`) .
- **[link]...[/link]** – tato dvojice nahrazuje počáteční a koncový tag odkazu na dokument (používá se například při formátování výstupu klasifikátorů – `[link]Dokument[/link]`).
- **[Text]** – obsah dokumentu (text).
- **[icon]** – vloží ikonu příslušející danému dokumentu.
- **[num]** – číslo dokumentu (použitelné například při hledání chyb v kolekci).
- **[Parent(All'_'):_]** – umožňuje odkazovat atributy všech nadřízených (rodičovských) částí dokumentu v rámci hierarchie.
- **[Parent(Top):_]** – odkazuje rodičovskou část dokumentu nejvyšší úrovně (tj. stojící v kořeni hierarchie).

1.5.8.2 Makra při formátování

Makra naplňují základní myšlenku univerzálnosti prezentace kolekcí a jejich obsahu. Pomocí maker je nadefinována naprostá většina výstupu systému Greenstone a právě jejich úpravou lze dosáhnout požadovaných výsledků. Spíše než při formátování zobrazení jednotlivých dokumentů hraji makra roli v celkovém vzhledu kolekce – ovlivňují umístění a podobu navigačních panelů, hlaviček dokumentů a podobně – vše v závislosti na nastavení prostředí (jazyk, mód zobrazení kolekcí ...). Při formátování vzhledu dokumentů se však makra využívají také – hlavně jako zástupné symboly pro proměnné prostředí. Zkušenější uživatel také může pomocí maker vytvářet jazykově závislé popisky (aby se například u pole s názvem objevilo při prohlížení s nastavenou češtinou „Název:“ a s nastavenou angličtinou „Title:“). V následujícím seznamu jsou uvedeny dva příklady maker, použitelných při formátování:

- `_httpprefix_` – zastupuje adresář, v němž je umístěn systém Greenstone
- `_httpcollection_` – adresář, v němž je umístěna kolekce

1.5.8.3 Podmíněné formátování

Dosud uvedené prostředky umožňují relativně bohaté formátování zobrazení jak u klasifikátorů tak i u dokumentů samotných, ovšem za předpokladu, že všechny dokumenty v kolekci (resp. odkazy na ně zobrazené v klasifikátorech), chceme formátovat stejně. Většinou však kolekce obsahují dokumenty různých formátů (text, grafika, video, zvuk ...), které je třeba zobrazovat různým způsobem. Právě na tento požadavek reaguje podmíněné formátování. Speciální řetězce uvnitř definice formátování umožňují na základě atributů dokumentu dále přizpůsobovat podobu výstupu. Základní podmíněný formátovací řetězec je:

```
{If}{podminka,formát pokud podmínka platí,
              formát pokud podmínka neplatí}
```

Na základě platnosti podmínky se vybere jeden z formátovacích řetězců. Podmínkou většinou bývá hodnota MD elementu. Následující příklad ukazuje formátovací řetězec, který na základě (ne)existence hodnoty elementu `Image` bud vykreslí obrázek nebo vypíše text:

```
{if}{[Image],<img src="_httpcollection_/index/
              [assocfilepath]/[Image]">,[text]}
```

Následuje podmíněné formátování, které slouží převážně pro výběr jedné z existujících položek. Při generování výstupu se postupně procházejí zleva doprava proměnné tak dlouho, dokud jedna z nich neplatí (odkazuje hodnotu) – tato proměnná se pak použije pro vytvoření výstupu:

```
{Or}{proměnná, proměnná, ..., řetezec}
```

Toto podmíněné formátovaní můžeme využít v případech, že u dokumentů nejsou definovány stejné elementy, ale požadovaná informace může být uložena v různých atributech.

Následující příklad vypíše buď hodnotu elementu `dc.Title`, pokud neexistuje tak hodnotu elementu `Title` a pokud ani tento atribut není definován, pak řetězec „Unknown“:

```
{Or}{[dc.Title],[Title],Unknown}
```

1.5.8.4 Příklad formátování

Na závěr této kapitoly bude uveden krátký příklad formátovacího řetězce. Jedná se o popis formátování klasifikátoru, určeného pro procházení obrázků v rámci kolekce. Číslování řádků bylo přidáno kvůli přehlednosti, není součástí formátování:

```
1. <td valign=top>
2.  [link]
3.  {If}{[dc.Thumbnail],
4.  <img src="_httpcollection_/index/assoc/[assocfilepath]/
5.                          [dc.Thumbnail]" alt="[dc.Title]">,
6.  {Or}{[srcicon],
7.     <img src="_httpcollection_/images/hiericon.gif"
                                  border="0">}}[/link]
8. </td>
9. <td valign=top>
10. [highlight]{Or}{[dc.Title],[Title],Untitled}[/highlight]
11. {If}{[Source],<br><i>([Source])</i>}
12.</td>
```

Klasifikátor zobrazuje do skupin hierarchicky roztříděné obrázky jako tabulku o dvou sloupcích (definice 1. sloupce odpovídá řádkům 1-8, definice druhého sloupce řádkům 9-12). Na úrovni skupin jsou zobrazeny zástupné ikony, určující, že se jedná o organizační položku hierarchie spolu s názvy skupin. Na úrovní dokumentů je každý obrázek zastoupen svou miniaturou a názvem, případně zdrojem, ze kterého pochází.

V prvním sloupci bude jako odkaz na dokument (proměnné `[link]` a `[/link]` na řádcích 2 a 7) sloužit buď miniatura obrázku (byla-li přiřazena v rámci atributu `dc.Thumbnail`) a nebo za běhu vytvořená miniatura (odkazovaná pomocí `[srcicon]`). Podmínka na řádcích 6 a 7 rozlišuje mezi zobrazením zástupné ikony hierarchie a zobrazením miniatury obrázku. V případě že není definována ani `dc.Thumbnail` ani `srcicon`, nejedná se o obrázek, ale o zástupce skupiny v hierarchii a v tomto případě zobrazíme ikonu hierarchie.

Ve druhém sloupci pak u každého dokumentu vypíšeme buď `dc.Title` nebo `Title`. Není-li ani jedna z položek definována, neznáme název dokumentu (skupiny dokumentů) a vypíšeme text „Unknown“ (viz řádek 10). Pod název se dále u dokumentů, u nichž je definován, vypisuje i zdroj, ze kterého pocházejí (viz řádek 11).

Podobným způsobem je možné nadefinovat i vzhled ostatních klasifikátorů a zobrazení jednotlivých dokumentů. Librarian jednak umožňuje pohodlnou orientaci a editaci formátovacích řetězců a také se stará o jejich správné uložení do konfiguračního souboru kolekce. U většiny nastavení pak platí, že je lze měnit „za běhu“ – po jejich uložení a obnovení stránek digitální knihovny se změny okamžitě projeví.

1.5.9 Překlady textů

Pod položkou *Translate text* se skrývá již na začátků zmíněná možnost překladu některých textových fragmentů. Jedná se zvláště o atributy kolekce (název, popis) a názvy indexů. V okně kromě seznamu fragmentů (položek), které je možné přeložit, jsou také panely popisující překlady přiřazené aktuální položce a umožňující tyto překlady editovat. K vybraným fragmentům lze přiřadit překlady do téměř libovolného používaného jazyka: od angličtiny, španělštiny, francouzštiny a ruštiny přes češtinu až po tak exotické jako je arabština a maorština. I přes velké množství jazyků zůstává možnost překladů silně omezena jen na vybrané části textu kolekce. Pokud bychom chtěli vytvořit skutečně „plně“ vícejazyčnou kolekci, museli bychom začít definovat jazykově podmíněná makra.

1.5.10 Přehled použitých MDS

Poslední položkou na záložce Design je položka *Metadata sets*. Slouží pouze pro přehled metadatových schémat použitých pro popis dokumentů kolekce. Ke každému MDS vypíše popis, seznam atributů s jejich popisem a omezeními na ně kladenými.

1.6 Vybudování kolekce – záložka Create

Budování kolekce je posledním krokem, který následuje poté, co upravíme veškerá potřebná nastavení. Proces vytváření kolekce má dvě části: import dokumentů (*import*) a výstavbu kolekce (*build*). Při importu jsou všechny dříve určené dokumenty postupně zpracovány pomocí pluginů, jsou k nim přiřazena metadata, případně asociovány další soubory a výsledek je ve speciálním formátu uložen k dalšímu zpracování. Ve fázi budování kolekce jsou z dat připravených při importu vytvořeny a zkomprimovány indexy a vytvořeny vazby na dokumenty. Proces výstavby je z větší části ovlivněn parametry přiřazenými kolekci (viz předchozí kapitoly), je však možné jej dále řídit pomocí nastavení na záložce *Create*. Záložka *Create* je rozdělena podobným způsobem jako ostatní záložky nástroje Librarian. V levé části se nachází panel sloužící pro výběr jedné ze tří položek:

- ***Import*** – otevírá nastavení dodatečných parametrů pro fázi importu.
- ***Build*** – otevírá nastavení dodatečných parametrů pro fázi budování kolekce.
- ***Message Log*** – zobrazuje záznamy o výsledcích importu a budování kolekce.

V pravé části záložky je umístěno okno, ve kterém se zobrazují detaily k jednotlivým položkám – nastavení parametrů a výpis záznamů. V dolní části záložky jsou pak umístěna tři tlačítka (*Build*, *Cancel* a *Preview Collection*), řídící proces budování kolekce a zobrazující výsledek. Po spuštění procesu vytváření kolekce se na záložce objeví okno, do kterého se vypisují hlášení generovaná v průběhu jednotlivých fází (pokud ovšem nepřesměrujeme výstup jinam). Na závěr je zobrazeno hlášení o výsledku procesu a v případě úspěšnosti nabídnuta možnost vytvořenou kolekci prohlížet. V následujících dvou tabulkách je uveden přehled parametrů ovlivňujících proces importu dokumentů a vytváření kolekce.

archivedir	určení místa, kde má být uložen výstup importu
collectdir	adresář, v němž je umístěna kolekce
debug	mají se výstupy importu vypisovat na standardní výstup?
faillog	kam se má ukládat záznam o chybách při importu
groupsize	kolik dokumentů bude popsáno jedním XML souborem
gzip	mají se komprimovat výstupní XML soubory?
importdir	kde je umístěn adresář obsahující soubory k importování
keepold	zachovávat dříve vytvořené importy?
maxdocs	maximální počet souborů, který je možné importovat
OIDType	způsob mají být generovány unikátní identifikátory dokumentů
OUT	kam vypisovat standardní výstup
removeold	komplementární s *keepold*
sortmeta	mají se údaje o dokumenty abecedně setřídit podle metadat?
statsfile	kam vypisovat statistiky importu
verbosity	určuje množství informací o importu poskytovaných při procesu
language	v jakém jazyce má proces importu poskytovat výstupy

tabulka 3: Dodatečné parametry pro import

archivedir	určení místa, kde jsou uloženy archivy (výstupy importu)
verbosity	určuje množství informací poskytovaných při procesu budování
builddir	kam má být uložen výstup (vytvořené indexy a asociované soubory)
maxdocs	maximální počet souborů, který bude zpracován
debug	vypisování výstupů na standardní výstup
mode	jaké kroky mají být při budování provedeny
index	které indexy mají být vytvářeny
keepold	zachovávat dříve vytvořené výstupy (staré indexy ...)
notext	neukládat komprimované texty – v rámci kolekce bude možné prohlížet jen originální dokumenty, ne texty z nich extrahované
allclassifications	prázdné klasifikace (tj. indexy pro procházení, které neobsahují žádná data) budou zachovány
create_images	systém se pokusí pro kolekci vytvořit zástupný obrázek (je nutné mít nainstalován program Gimp)
collectdir	adresář obsahující kolekci
out	určení souboru či handleru určených k vypisování výstupu
no_strip_html	speciální nastavení při použití indexování pomoci MGPP – HTML tagy v v indexovaném textu budou zachovány
faillog	jméno souboru, do něhož budou ukládána hlášení o chybách
language	v jakém jazyce mají být poskytovány informace o průběhu procesu

tabulka 4: dodatečné parametry pro budování kolekce

1.7 Modifikace nastavení kolekce

Pokud se nám podařilo přidat soubory k importu, přiřadit k nim metadata, nastavit potřebné parametry a kolekci vybudovat, můžeme nyní zhodnotit jak výsledek odpovídá našim požadavkům. Zvláště vytváříme-li zcela nové sbírky s unikátním formátováním, nepodaří se hned na první pokus dosáhnout potřebné funkčnosti a vzhledu. V takovém případě ovšem není problém patřičná nastavení upravit a kolekci vytvořit znovu. Obsahuje-li kolekce velké množství dokumentů, můžeme pomocí parametrů importu a budování omezit počet zpracovávaných souborů a na takto omezeném „výseku“ sbírky změny testovat.

Většinu změn v zobrazení obsahu kolekce můžeme provádět aniž bychom museli kolekci znovu budovat. Stačí jen upravit formátovací řetězce a uložit je do konfiguračního souboru. Po obnovení stránek digitální knihovny Greenstone se úpravy hned projeví.