

NÁVRH A REALIZACE WWW PREZENTACE ČKR

Šárka Ocelková

Ústav výpočetní techniky MU v Brně, Botanická 68a, 602 00 Brno, ČR
E-mail: ocelkova@ics.muni.cz

Abstrakt

U zrodu www prezentace České konference rektorů stál požadavek na jednoduchou modifikaci, levné řešení a jednotný design. Článek popisuje analýzu a realizaci této prezentace (<http://crc.muni.cz/>), jejíž řešení je postaveno na standardech XML, XSL a HTML s důsledným oddělením designu od vlastního obsahu stránek, přičemž obsahové změny provádí laický uživatel.

1. Jak to všechno začalo?

Přibližně před rokem se na tým tvůrců www stránek Masarykovy univerzity (dále jen tvůrce) obrátila Česká konference rektorů (dále jen zákazník) s potřebou mít také svoji vlastní www prezentaci. Zákazník měl pouze orientační představu o tom, jak by prezentace měla vypadat, jasná byla pouze hlavní témata, která by prezentace měla obsahovat. Co se pod nimi bude skrývat a jakého budou rozsahu, zdaleka nebylo odhadnutelné. Na celé zadání však bylo možné pohlížet jako na tři samostatné části, na kterých lze z počátku téměř nezávisle pracovat: 1. grafická podoba stránek; 2. technická realizace; 3. datový obsah stránek. Tato tři hlediska byla do kompetencí zúčastněných stran rozdělena následujícím způsobem:

- ad 1) grafiku navrhne a vytvoří tvůrce, zákazník ji ovšem musí schválit (jakoukoliv následnou změnu již schválené grafiky ze strany tvůrce musí zákazník opět vždy schválit);
- ad 2) technická realizace je čistě na tvůrci, který rovněž zodpovídá za trvalý a bezchybný chod www prezentace;
- ad 3) datový obsah je záležitostí zákazníka, sem patří i vlastní vzhled jednotlivých dokumentů prezentovaných na www stránkách.

O grafice, ač je na www stránkách právě to nejdůležitější, co dodává celé prezentaci tvář a určitý charakter, bohužel nelze po technické stránce mnoho říci. Vše je závislé na tom, zda se grafikovi mihne hlavou nějaká pěkná představa, a je-li navíc i počítačově gramotný a nápad hned převede do elektronické podoby, je to přímo ideální. Z několika návrhů byl nakonec vybrán ten, který dnes může kdokoliv spatřit na <http://crc.muni.cz/>.

Další rozhodující skutečností, která měla pro tvůrce nezanedbatelný vliv na technickou realizaci, byla aktualizace obsahu stránek. Jinými slovy řečeno, bude-li zákazník potřebovat jakoukoliv změnu ve vlastním obsahu www stránek (od obyčejného překlepu ve slově až po vystavení dalšího nového dokumentu), nebylo by únosné, aby pokaždé musel o doplnění/aktualizaci stránek žádat tvůrce. Musí tedy existovat jednoduchý způsob, jak si zákazník bude moci do obsahu stránek zasahovat sám, ale tak, aby nechtěně nedopatřením neohrozil celou prezentaci. K jakému řešení tvůrci nakonec dospěli, je obsahem dalších kapitol tohoto článku.

2. Jaké jsou možnosti řešení

Nejprve bylo třeba utvořit si představu o konečném rozsahu www prezentace. Po několika konzultacích se zákazníkem vyplynulo, že se bude jednat o web středního rozsahu – několik desítek www stránek – přičemž (vyjma úvodní stránky) budou všechny ostatní stránky ve stejném grafickém provedení. Z toho je také patrný postoj k možným řešením:

- a) všechny stránky psát „ručně“ přímo jako HTML (resp. SHTML)
 - v dnešní době možností nejrůznějších technologií by toto řešení bylo krajně neefektivní. Dostí závažná nevýhoda je opisování téhož kódu, který určuje vzhled stránek, do všech stránek. Tuto nevýhodu lze částečně obejít pomocí SHTML, tímto lze bohužel ošetřit pouze „obal“ (jednotné záhlaví a zápatí) stránky, ne však vnitřní vzhledové a formátovací prvky. Tato varianta je nevýhodná i pro zákazníka. Bude-li sám provádět obsahové změny ve stránkách, nejen že by musel znát jazyk HTML, ale hrozí zde reálné nebezpečí neúmyslného překlepu a tím i (byť dočasné) grafické znehodnocení stránky. Proto také byla tato možnost téměř okamžitě vyloučena.
- b) oddělit vlastní data od HTML kódu formátujícího stránky
 - takové řešení nabízí možnost zabránit zákazníkovi přístup do části určující vzhled a formátování stránek a tím eliminovat možné grafické znehodnocení v případě překlepu. Pro vlastní uložení dat se nabízí buď relační databáze nebo soubor s předem definovanou strukturou dat (nejvhodnější se vzhledem k možnostem jevílo XML). Toto řešení ale musí být podpořeno nějakým softwarem, který bude schopen přistupovat k uloženým datům a přidáním formátovacích prvků vytvoří vlastní html stránku. Tento software může být navíc naprogramován i tak, že bude úspěšně detekovat vybrané případné chyby v datech.

Ze dvou uvedených variant byla vybrána ta „bezpečná“ z pohledu uživatele, následující text již objasní příslušnou volbu uložení dat a softwaru pro vytváření html stránek.

3. Databáze versus XML

Jak již bylo zmíněno výše, obě tato řešení umožňují oddělení technické realizace (za kterou odpovídá řešitel) od vlastních dat (za které odpovídá zákazník). Na rozhodnutí, zda jako úložiště dat použít relační databázi nebo XML soubor, má vliv podrobnější analýza předpokládané podoby www stránek. Ukažme si proto dva typické příklady: adresář a usnesení.

- a) adresář
 - je určitým způsobem strukturovaný seznam všech členů ČKR, jejich funkcí, kontaktů a vysoké školy na níž působí, včetně uvedení adresy. Ukázka adresáře je na obrázku 1.

| Adresář | |
|--|--|
| 1. Prof. Ing. Jan Novák, CSc., rektor Univerzita Karlova v Praze předseda České konference rektorů Václavské nám. 1 100 00 Praha 1 | tel.: 123 456 789 987 654 321 fax: 123 456 789 e-mail: jan.novak@abc.cuni.cz crc.president@muni.cz |
| 2. Prof. MUDr. PhDr. Jana Nováková, CSc., rektorka Univerzita Palackého v Olomouci Křížkovského 7 700 07 Olomouc | tel.: 666 555 444 333 222 111 fax: 999 888 777 987 987 987 e-mail: novakova@xyz.upol.cz |
| 3. atd. | |

Obr. 1: Ukázka adresáře

Data obsažená v adresáři se však budou objevovat i na řadě dalších stránek, např. seznamy členů předsednictva, pléna a komor – vše jsou ve skutečnosti jen určité podmnožiny seznamu všech členů (tj. adresáře). Dalším příkladem je připravovaná historie členů, kde se kromě jmen členů objeví opět názvy vysokých škol.

b) usnesení

– jsou textové dokumenty z jednotlivých zasedání ČKR, které mají všechny přibližně stejnou strukturu: záhlaví, jednotlivé body usnesení (mohou být i víceúrovňové) a zápatí. Jak takové usnesení vypadá ukazuje obrázek 2.

| Usnesení 50. zasedání České konference rektorů Praha, 19. 10. 2000 | |
|---|--|
| Česká konference rektorů (ČKR) přijala na svém 50. zasedání následující usnesení: | |
| 1. ČKR zhodnotila | |
| 2. a) | |
| b) | |
| 3. | |
| V Praze dne 19. října 2000 | Za Českou konferenci rektorů Prof. Ing. Jan Novák, CSc. předseda |

Obr. 2: Ukázka usnesení

Informace ze záhlaví (ze kterého zasedání, místo a datum) však slouží také jako podklady k další stránce – přehled usnesení. Její ukázka je na obrázku 3.

| Přehled usnesení | |
|------------------|--|
| | |
| 50. zasedání ČKR | (Praha, 19. října 2000) |
| 49. zasedání ČKR | (Lednice na Moravě, 21.-22. září 2000) |
| 48. zasedání ČKR | (Opava, 25.-26. května 2000) |
| 47. zasedání ČKR | (Brno, 24.-25. února 2000) |
| | |

Obr. 3: Ukázka přehledu usnesení

Jak naznačují příklady, různé www stránky mohou obsahovat tatáž data. Odtud plyne, že na datové obsahy stránek nelze pohlížet jako na samostatné množiny dat, ale je nutné zohlednit v datové analýze právě již výše zmíněnou skutečnost vícenásobného výskytu týchž dat. Základní položky, které je nutné evidovat pro naše dva příklady, jsou:

a) adresář

– budou potřeba evidence členů, kontaktů, funkcí a vysokých škol. Ke každému členu bude možnost evidovat jméno, příjmení, tituly, kontakty, funkce a vysoká škola. Z obrázku 1 je patrné, že k jednomu členu může existovat více kontaktů, tj. telefonů, faxů nebo e-mailových adres. U vysokých škol bude kromě názvu uvedena také adresa, případně URL www prezentace. V případě funkcí stačí pouze jejich název.

b) usnesení

– u každého usnesení bude potřeba evidovat pořadí (které je dáno pořadím zasedání, na němž bylo usnesení přijato), místo a datum konání, podpis a vlastní text usnesení, který může a nemusí být dále strukturován.

Nyní si ukažme, jak by vypadala struktura dat v případě uložení do relační databáze a do XML souboru.

3.1 Struktura dat v relační databázi

a) adresář

– pro každou výše uvedenou evidenci (v databázové terminologii entitu) je rozumné uvažovat samostatnou tabulku, tj. tabulku pro členy, školy a funkce. Každá entita musí mít svůj jednoznačný identifikátor (primární klíč) id. Je-li pak potřeba u členu uvést vysokou školu, je v entitě člen uvedeno pouze id_školy (cizí klíč do tabulky škola). Zcela obdobně je to s uvedením funkce. Zdánlivý problém může nastat s evidencí kontaktů. Jelikož je vztah mezi členem a kontaktem obecně 1:N, měla by ve správném návrhu databáze existovat další samostatná tabulka s kontakty, kde by např. nějaký atribut „typ“ říkal, zda se jedná o telefon, fax nebo e-mail a každý kontakt by byl vždy vázán na konkrétního členu uvedením id_člena. Tuto situaci je možné bez újmy na obecnosti zjednodušit tak, že u každého členu budou uvedeny atributy telefon, fax a e-mail, v případě vícenásobného výskytu budou jednotlivé kontakty odděleny např. čárkou. Strukturu databáze ukazuje obrázek 4.

| |
|--|
| <pre>člen (id, titul_před_jménem, jméno, příjmení, titul_za_jménem, id_funkce_v_ČKR, komora, id_školy, funkce_ve_škole, telefon, fax, e-mail) škola (id, název, url_www_stránek, adresa_ulice, adresa_číslo, adresa_PSČ, adresa_město) funkce (id, název)</pre> |
|--|

Obr. 4: Databázová struktura adresáře

b) usnesení

– na první pohled se může zdát, že usnesení je jediná entita, jejíž identifikátor id může být v podstatě pořadí zasedání, na kterém bylo usnesení přijato. Další položky jsou místo a datum konání (vztahuje se na zasedání) a konečně v jednom dalším atributu by mohl být uložen celý text usnesení již naformátovaný HTML značkami. Pro zákazníka by ale toto řešení nebylo příjemné, neboť by se musel naučit jazyk HTML. Protože mají jednotlivá usnesení poměrně jednoduchou strukturu (jednotlivé body a podbody), lze tyto části uložit do samostatné tabulky. Musí být ale kromě identifikátoru usnesení, k němuž se část vztahuje, nutně uvedeno také pořadí části, její úroveň a typ číslování (číslem, písmenem nebo odrážkou). Jak by mohla struktura vypadat ukazuje obrázek 5.

```

usnesení (id (=pořadí_zasedání), místo_konání, datum_konání)
usnesení_část (id_části, id_usnesení, pořadí_části, úroveň,
               typ_číslování, text_části)

```

Obr. 5: Databázová struktura usnesení

3.2 Struktura dat v XML souboru

a) adresář

– obdobou databáze mohou být jeden nebo více XML souborů. V případě adresáře lze každou uvažovanou tabulku z předchozí kapitoly reprezentovat jedním souborem: člen, škola a funkce. Každé databázové entitě pak odpovídá hlavní element souboru: <ČLEN>, <ŠKOLA> a <FUNKCE>. Atributy tabulek odpovídají atributům hlavního elementu. Oproti databázi lze však pomocí XML velmi snadno řešit problém vícenásobných kontaktů – součástí každého elementu <ČLEN> budou vnořené značky <TELEFON>, <FAX> a <E-MAIL>, které budou u každého člena tolikrát, kolik se u něho eviduje telefonů, faxů nebo e-mailů. Struktura hlavních XML elementů je ukázána na obrázku 6.

```

<ČLEN id="..." titul_před_jménem="..." jméno="..." příjmení="..."
      titul_za_jménem="..." id_funkce_v_ČKR="..." komora="..."
      id_školy="..." funkce_ve_škole="...">
  <TELEFON>... ..</TELEFON> <TELEFON>... ..</TELEFON> ... ..
  <FAX>... ..</FAX> <FAX>... ..</FAX> ... ..
  <E-MAIL>... ..</E-MAIL> <E-MAIL>... ..</E-MAIL> ... ..
</ČLEN>

<ŠKOLA id="..." název="..." url_www_stránek="..." adresa_ulice="..."
      adresa_číslo="..." adresa_PSČ="..." adresa_město="..."/>

<FUNKCE id="..." název="..."/>

```

Obr. 6: Struktura hlavních elementů v XML souborech pro adresář

b) usnesení

– databázovou tabulku je opět možné jednoduše reprezentovat souborem usnesení s hlavním elementem <USNESENÍ>. Díky strukturovatelnosti XML je zde ovšem mnohem lepší možnost uložení vlastního textu usnesení, členěného na jednotlivé body a podbody. Příslušným vnořováním značek <BOD> (eventuálně <TEXT> pro vlastní text bodu) lze elegantně docílit toho, co v databázi jen velmi obtížně další tabulkou. Strukturu usnesení v XML ukazuje obrázek 7.

```

<USNESENÍ pořadí="..." místo_konání="..." datum_konání="...">
  <BOD typ_číslování="...">
    <TEXT>... ..</TEXT>
    <BOD typ_číslování="...">
      <TEXT>... ..</TEXT>
    </BOD>
    ... ..
  </BOD>
  <BOD> ... .. </BOD>
  ... ..

  <ZÁPATÍ>
  <DATUM>... ..</DATUM>
  <PODPIS>... ..</PODPIS>
  </ZÁPATÍ>
</USNESENÍ>

```

Obr. 7: Struktura hlavního elementu v XML souboru pro usnesení

3.3 Proč bylo vybráno XML

Aby mohl zákazník svá data libovolně modifikovat, aniž by musel stále žádat tvůrce stránek, je třeba zřídit mu k nim přístup. Zpřístupnění relační databáze je komplikovanější z hlediska nároků na klientský počítač, neboť je potřeba určitý software pro přístup do databáze. Pro zpřístupnění XML souborů s daty postačuje pouze vytvoření účtu zákazníkovi na www serveru a zpřístupnění části disku, kde jsou uložena jeho data. K editaci XML souboru postačuje obyčejný textový editor.

Pro variantu XML také hovoří možnost jeho strukturování, jak ukazují výše uvedené příklady. Nevýhodou naopak je nutnost zaškolení zákazníka do formátu zápisu dat. Může se zdát, že v případě použití relační databáze by tento problém odpadl, neboť zápis dat je vlastně jen pouhé vyplňování tabulek. Složitě strukturované texty, kterých je v ČR hodně (např. zápisy ze zasedání, výroční zprávy apod.), by se ale bez speciálně navržených podpůrných formulářů vkládaly do databáze jen velmi obtížně.

Po celkovém zhodnocení tedy nakonec zvítězila varianta XML díky možnosti strukturování dat a minimálním softwarovým nárokům.

4. Vnitřní architektura

4.1 Vygenerování www stránek

Nyní je již k dispozici sada XML souborů, v nichž jsou uložena potřebná data. Poté byla ke každému typu HTML stránky naprogramována XSL transformace (šablona), která udává předpis, jak se mají konkrétní data v XML převést do HTML podoby. Z toho pak vyplynul další úkol – zajistit automatické vygenerování celé www prezentace. Pro vytvoření každé html stránky je nutné vědět, z jakých dat (xml souboru) má vzniknout, kterou šablonou mají

být data transformována a kam se má výsledek transformace uložit (tj. jméno a umístění výsledného souboru, který již bude součástí www prezentace). Z těchto důvodů vznikl konfigurační soubor (rovněž ve formátu XML), kde jsou tyto informace popsány. Aby se při uvádění souborů nemusela neustále opakovat celá cesta k souborům, jsou v konfiguračním souboru navíc dodány atributy `xml-root-dir`, `xsl-root-dir` a `html-root-dir`, v nichž je uložena společná část cesty (jednotlivě pro xml zdroje, xsl transformace a html cíle) od kořene www serveru. Vše je ukázáno na obrázku 8.

```
<FILES xml-root-dir="... cesta k adresáři s XML zdroji ... "
      xsl-root-dir="... cesta k adresáři s XSL transformacemi ... "
      html-root-dir="... cesta ke kořenu www serveru crc.muni.cz ... ">

<ITEM xsl="adresar.xsl" xml="adresar.xml" html="directory/index.html"/>
... ..
<ITEM xsl="usneseni_index.xsl" xml="usneseni.xml"
      html="resolution/index.html"/>
<ITEM xsl="usneseni.xsl" xml="usneseni.xml" html="resolutions/65.html"/>
<ITEM xsl="usneseni.xsl" xml="usneseni.xml" html="resolutions/66.html"/>
... ..
<ITEM xsl="aktuality.xsl" xml="aktuality.xml" html="news/index.html"/>
<ITEM xsl="kalendar.xsl" xml="kalendar.xml" html="calendar/index.html"/>
... ..
<FILES>
```

Obr. 8: Ukázka konfiguračního souboru

Toto řešení má zatím ještě jeden nedostatek, který není nijak závažný a navíc je řešitelný (lze doprogramovat). Slouží-li jeden XML soubor jako zdroj dat pro několik html souborů (viz příklad usnesení, kde jsou všechna data o usneseních v jediném souboru a následně se z nich generují samostatné www stránky jednotlivých usnesení), musí být v konfiguračním souboru uveden tolikrát, kolik z něj má být vygenerováno html souborů. Tento nedostatek je však pozitivně využít, neboť cílové jméno html souboru slouží jako parametr pro xsl transformaci (např. vygeneruje se stránka pro jedno konkrétní usnesení).

Pro zpracování xsl transformací dle konfiguračního souboru byly použity knihovny SAXON (provádění transformací) a XALAN (parsování XML).

Při požadavku na vygenerování stránek se vždy vygeneruje celá sada stránek. Není zde řešeno částečné přegenerování (zvolených) stránek pro případ, kdy je potřeba uplatnit změnu jen na jedné či několika málo stránkách. Zatím je však toto řešení dostatečné, neboť vzhledem k rozsahu webu trvá přegenerování celé prezentace jen několik desítek sekund.

4.2 Jednotný vzhled stránek

Kapitola 2 pojednávala o možnostech technické realizace a také o volbě řešení, které odděluje vlastní data od html kódu formátujícího stránky, jenž pak může být uložen centrálně na jediném místě. Zbývá proto objasnit, jak je řešen jednotný vzhled stránek.

V sadě XSL transformací je speciální obecná transformace, která se využívá při generování všech stránek. Jedná se o soubor obecných pravidel a funkcí (např. pro výpis jednotného záhlaví a zápatí www stránky, nadpisů různých úrovní, odkazu „o úroveň výš“ ve stránkové

struktury apod.) zpřístupněný všem ostatním transformacím, které tak mohou tato pravidla funkce libovolně využívat nebo se na ně odkazovat.

4.3 Vývoj webu a transakční přegenerování

Jelikož www prezentace ČKR neustále „žije“ (dle potřeby se obměňují stránky aktualit nebo kalendáře, v současné době se připravuje historie atd.), je potřeba mít možnost podívat se, jak bude po provedení změn vypadat výsledná www stránka. Je také nutné počítat s případným překlepem či jinou chybou ve zdrojových datech nebo vyvíjených transformacích, což může zapříčinit buď nemožnost uplatnění transformace a tím zhavarování programu pro generování, nebo zveřejnění překlepu ve výsledném souboru prezentace. Obojí je samozřejmě nežádoucí a u veřejné www prezentace nelze takto riskovat. Proto byla zvolena (jak se tvůrcům již osvědčilo v jiných projektech) varianta dvou www serverů (služeb), z nichž jeden je vývojový a druhý ostrý. Oba mají identickou adresářovou strukturu, tj. všechny soubory s daty, transformacemi i vygenerovaná prezentace existují dvakrát na dvou oddělených místech. Na vývojovém serveru je možné soubory s xml daty a xsl transformacemi modifikovat a generovat z nich pokusně výsledné www stránky, na druhém, ostrém serveru, nejsou určeny k modifikování, ale pouze k nahrazování z otestované varianty vývojové části.

Přegenerování www prezentace má ještě jednu „pojistku“ proti vzniku chyb (nelze jí ošetřit vlastní překlepy v textu, ale koncepční chybu v XML nebo XSL). Ze zkušeností s vývojem a provozem oficiální www prezentace MU v Brně byla do programu, který generuje www stránky, implementována vlastnost transakčního přegenerování, které zajistí, že v případě chyby v tomto procesu zůstává zveřejněna stará verze stránek. Toto řešení je založeno na skutečnosti, že se celá www prezentace nachází na UNIX serveru a je tak možné využít vlastnosti symbolických linků. O této problematice podrobněji pojednává [1].

5. Rozhraní pro zákazníka

Zákazník byl úspěšně zaškolen do formátu zápisu dat v XML a do logiky přegenerování www stránek, kterou pečlivě dodržuje. Pro editaci dat mu byl doporučen jednoduchý textový editor se zvýrazňováním syntaxe, spuštěním jednoho příkazu pak může přegenerovat vývojový web. S přenosem na ostrý web to již tak snadné není, neboť na vývojovém webu mohou být některé soubory rozpracovány. Z tohoto důvodu není možné nabídnout zákazníkovi jediný příkaz, který by najednou překopíroval všechny datové zdroje na ostrý web a přegeneroval www stránky. Proto byla vytvořena a zákazníkovi předložena sada vhodně nazvaných příkazů, které překopírují na ostrý web vždy jen ty datové zdroje, které se týkají určité samostatné části www prezentace, a následně přegenerují všechny www stránky na ostrém webu.

6. Závěr

Všechny www prezentace, které tvůrci až doposud vytvořili, vždy vytvářeli s vědomím, že s jejich „zákulisím“ budou operovat jen odborníci v oboru. Nebylo tedy třeba zohledňovat laický přístup. Právě proto zakázka na www prezentaci ČKR přinesla cenné zkušenosti s laickým zákazníkem.

Výsledné technické řešení má sice ještě drobné nedostatky (popsány výše v textu), ty však v současné době nejsou ke škodě. Plánuje se vylepšení přegenerování jen na vybrané www

stránky a dořešení XML struktury ještě některých datových zdrojů stránek. V současné době se také připravuje historie členů ČKR.

Za zmínku také stojí skutečnost, že se k výročí 30. ročníku konference SOFSEM připravuje elektronická podoba všech sborníků založená právě na technologii XML a XSL.

Literatura:

1. Ocelka, Jaromír. WWW Server v přílivu uživatelů Internetu. In Tvorba softwaru 2003. Vyd. první 2003. Ostrava: TANGER, s.r.o., 2003. ISBN 80-85988-83-6, s. 154-160
2. Harold, Eliotte Rusty. XML Bible, 2nd Edition. 2001, ISBN 0-7645-4760-7, 1206 stran
3. Výzkumný záměr ÚVT MU: Digitální knihovny. URL:
http://wwwwdata.muni.cz/research/cez_item.asp?ID=CEZ:J07/98143300004