

IMPLEMENTACE MODERNÍCH METOD PŘI TVORBĚ INTERNETOVÉHO PORTÁLU

MODERN METHODS IMPLEMENTATION USED BY CREATION OF THE INTERNET PORTAL

Pavel Šimek, Vratislav Čermák

Souhrn

Programové řešení portálu by mělo umožňovat práci s různými datovými formáty (především XML), podporovat zpřístupnění informací na různých platformách a dovolovat tvorbu modulárního a opakovatelně využitelného kódu. Konkrétní implementace těchto přístupů je prezentována na nové verzi portálu Agris. Je založena na přednostech skriptovacího jazyka PHP 4.3, jazyku XML, XHTML, XSLT a databázového serveru MS SQL Server 2000. Nový koncept se vyznačuje především přívětivějším uživatelským prostředím a také univerzálností a modularitou programového řešení. Příspěvek se zabývá především aspekty vývoje portálového řešení a získanými poznatky.

Summary

Portal software should be suitable for various data formats (especially XML), to support access to information on various platforms, and to allow the creation of a modular, reusable code. The new version of Agris portal, based on the advantages of PHP 4.3 scripting language, XML, XHTML and XSLT languages and a Microsoft SQL Server 2000 database server, is presented as an example of such approach implementation. User-friendly environment and universal and modular software are the features of this new concept. This paper deals primarily with the aspects of portal designing and the experience gained.

Key Words

portal, Agris, agriculture, PHP, XML, XSLT, XHTML, SQL, object, class

Klíčová slova

portál, Agris, zemědělství, PHP, XML, XSLT, XHTML, SQL, objekt, třída

Úvod

Téměř na všech stránkách internetových e-zinů je portál označován jako nejvhodnější nástroj pro zpřístupnění informací určité skupině uživatelů. Portál je vlastně vstupní brána k informacím. Z technologického hlediska může být takovýmto „portálem“ webová aplikace komunikující s různými aplikacemi skrze různá rozhraní, napojená na databázové stroje a to samozřejmě s plným využitím možností moderních skriptovacích jazyků. Portálem může být také nazýván soubor provázaných statických www stránek, který obsahuje včasné zobrazení všech informací. Údržba staticky řešeného portálu je z časového hlediska velice náročná a dlouhodobě nerealizovatelná. Proto se tvůrci všech rozsáhlejších aplikací takovémuto přístupu vyhýbají.

Cíl práce a metodika

Vývoj agrárního WWW portálu Agris byl zahájen v polovině roku 1999 a do své finální podoby se dostal začátkem roku 2000, ve kterém také získal ocenění Zlatý klas na nejstarší a nejnavštěvovanější zemědělské výstavě v České republice Země živitelka. Na svou dobu byly použity moderní WWW technologie a databázový server MS SQL Server 2000. V roce 2002 se však díky neúprosnému pokroku v oblasti ICT stala jeho koncepce částečně zastaralá a z programového hlediska hůře udržitelná, a proto v září téhož roku byl započat vývoj nové verze.

Pokud má portál pracovat efektivně a má být samozřejmě co možná nejvíce přizpůsobitelný a rozšiřitelný, musí programovému řešení předcházet dostatečná analýza, měl by být řešen objektivně a měl by využívat nejmodernější technologie v oblasti tvorby webu, které zajistí uvedené požadavky.

Moderní technologie na straně klienta (DHTML, ActiveX, JavaScript, Java, CSS, apod.) jsou však dnes základním úskalím portálů pravidelných návštěvníků neboli veřejných portálů. Tvůrci takového portálu musí brát v potaz nejednotnost ve verzi a značce prohlížeče, se kterým si budou návštěvníci web prohlížet.

OOP při tvorbě portálu

Při použití objektově orientovaného přístupu lze rozdělit vyvíjený portál na jednotlivé objekty jako celky. Každý problém je řešen obecně a vzniká tak knihovna objektů, o které lze předpokládat, že bude v budoucnu využita i při tvorbě jiných aplikací.

Nové objekty je žádoucí vytvářet z objektů již existujících s využitím vazeb skládání, dědičnosti, závislosti a delegování a nově implementovat jen ty charakteristiky, jimiž se liší od stávajících objektů. OOP zatím překonává nejlépe ze všech známých přístupů nebezpečí, že tvůrci portálů nebo jiných systémů vyčerpají svoji energii na jednotlivých částech, vynucených obtížnou implementací detailů.

U dobře navrhnutého objektově orientovaného portálu je možné skrýt značnou část primitivních operací, které jsou při tvorbě na obtíž. Jsou totiž zapouzdřeny do kódů metod objektů a takové objekty potom spolu komunikují na vyšší úrovni.

Mnoho objektů lze využít jako hotové komponenty v knihovně systému. Celý portál může být navrhován, testován a odladěn po jednotlivých částech, objektech.

Stádium expanze

Stádiem expanze se označují vývojové fáze portálu zadání a analýza. Zadání vcelku není tak složité jako samotná analýza, která by se měla skládat ze čtyř samostatných částí:

1. analýza zdrojů
2. analýza procesů
3. datová analýza
4. analýza uživatelského prostředí

Analýza zdrojů by měla být provedena jak na straně uživatelů, administrátorů (zadavatele) tak i na straně zhotovitele samotného portálu. Zadavatel portálu by měl analyzovat dostupné informační zdroje, místa, kde informace vznikají a požadovanou formu v jaké by měly být portálem zpracovávány a prezentovány. Analýzou zdrojů na straně uživatele se míní analýza požadavků na systém. Obě tyto analýzy zpracovává zadavatel. Analýza zdrojů na straně zhotovitele portálu představuje především plánování použitých hardwarových a softwarových prostředků, které má zhotovitel k dispozici, které jsou pro daný účel vhodné, a pro které lze efektivně využít projektové řízení.

Vnitřní transformace informací uvnitř portálu, způsoby zobrazení, použité nástroje a systém získávání informací by měly tvořit základ analýzy procesů. Tato analýza vyúsťuje v model procesů. Na jedné straně vzniká modelovaný systém, který je tvořen řadou procesů, na druhé straně je tedy i odpovídající funkční model obsahující množinu vzájemně souvisejících procesů. Na základě této analýzy je naprogramována aplikační logika celého systému.

Následně je možné přistoupit k analýze datové základny, tedy databází, ve kterých budou uchovávána veškerá data neboli informace. Datová analýza musí vycházet z předcházejících dvou kroků. Kvalitně navržená datová základna se projeví především v postimplementačním období, tedy v době, kdy zadavatel začne vznášet dodatečné a nové požadavky na funkcionalitu portálu. V případě bezchybně navržené datové základny a využití OOP by měla být pracnost těchto změn minimální.

Čtvrtá část základní analýzy se týká především uživatelů, kteří budou se systémem pracovat. Je zapotřebí zjistit v jakém prostředí jsou zvyklí pracovat, jak jsou zde rozmístěny ovládací prvky, jak se systém chová, jak vypadá a jaká je úroveň informační gramotnosti. Kromě toho by měla analýza přinést informace o tom, co by systém měl umět a jaké funkce by v něm měly být zahrnuty. Takovýto způsob analýzy je možný především u uzavřených portálů, kde je cílová skupina známa. U portálů pravidelných návštěvníků se tato analýza provádí především prostřednictvím analýzy fungování podobných a již osvědčených systémů na internetu.

Stádium konsolidace

Fáze implementace, ladění, testování, provoz a údržba se označují stádiem konsolidace proto, že v těchto etapách se model, který je produktem předchozího stádia expanze, postupně stává fungující aplikací. Nejobtížnější fází je fáze implementace, kde se po návrhu a realizaci designu či vzhledu přejde k aplikační logice celého portálu.

Design

Na základě stanovení a návrhu designu je možné řešit ovládání tak, aby bylo jednotné v celém prostředí. Na úrovni kódu je nutné vytvořit tzv. stylesheet (katalog použitých stylů) a stanovit způsoby jeho používání a rozšiřování.

Stylesheet je v prostředí internetového portálu tvořen pomocí kaskádních stylů (CSS). Jako nejrozměšnější se jeví zmíněný stylesheet uložit do externího souboru. Pokud budou všichni vývojáři využívat stylesheet podle pravidel je možné jednoduchým způsobem a během krátké doby měnit vzhled celého portálu. I zde je potřeba mít na mysli, že dodatečná implementace kaskádních stylů je časově velice náročná.

Aplikační logika

Nejdůležitější část pro provoz portálu představuje funkční jádro www aplikace. V naprosté většině případů se ale nejedná o časově nejnáročnější část projektu. Mnohem náročnější na čas je tedy samotná analýza, jejíž kvalitní provedení snižuje časovou náročnost realizace aplikační části.

Výše uvedené skutečnosti otevírají cestu efektivnímu využití skriptovacích jazyků a technologií na straně serveru. Mezi nejpoužívanější patří ASP 3.0 nebo ASP.NET s využitím jazyků Visual Basic a C#, PHP 4 a Java Server Pages. V současné době již mají implementovanu podporu objektového programování všechny zmiňované skriptovací jazyky na dobré úrovni a umožňují tak vytvářet webovou aplikaci efektivně.

Programováním na úrovni tříd s využitím inheritance je možné oddělit designovou a aplikační část webu, což u skriptovacích jazyků dřívější generace bylo velice obtížné. Pokud je vytvořen kvalitní objektový návrh a metody jsou navrženy dostatečně univerzálně, je možné jednotlivé stránky portálu zkonstruovat pouze vytvořením z několika instancí tříd.

Technologie Microsoft .NET má především velkou přednost v již vytvořené rozsáhlé objektové knihovně tříd. Celý portál je však třeba vytvořit na základě výše uvedených analýz. Proto není pro kompletní tvorbu nového portálu příliš důležité jaká platforma bude zvolena, ale spíše na tvorbě kvalitního objektového návrhu.

XML, XHTML

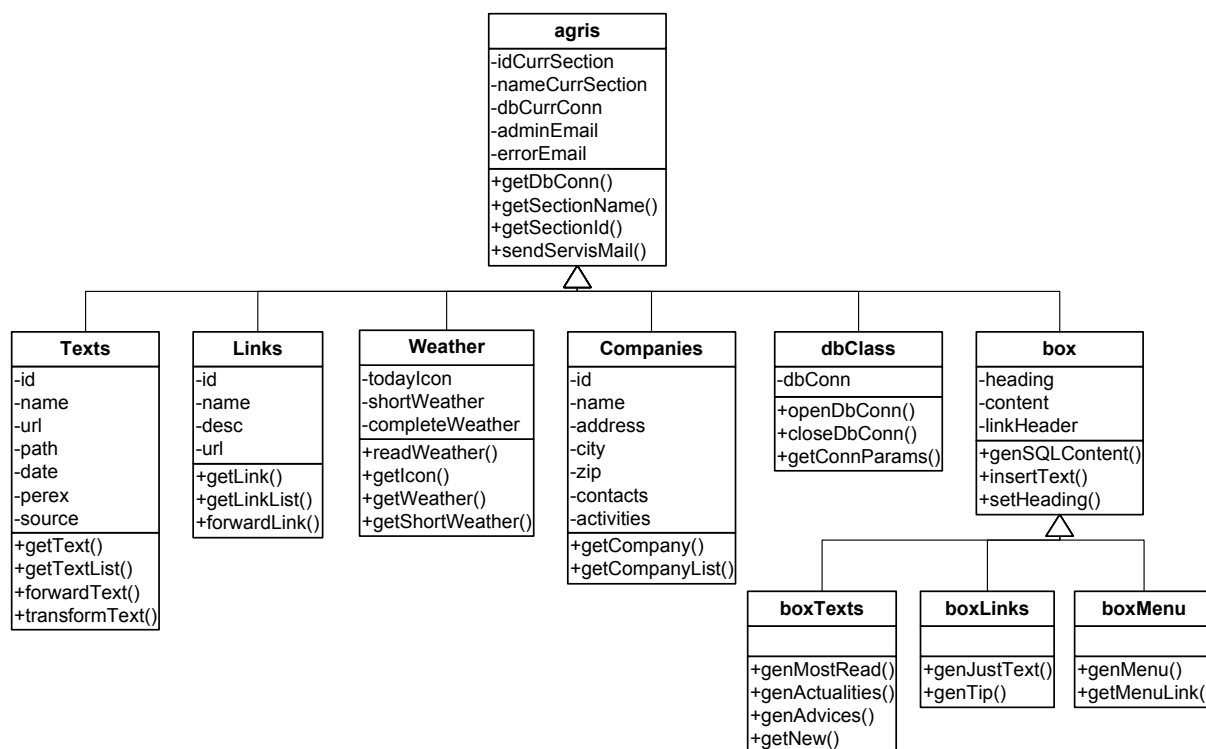
Každý portál by svá rozsáhlá data měl uchovávat v určitém, snadno rozšiřitelném, standardu, který lze snadno distribuovat do různých koncových zobrazení. Jako nejlepší se jeví formát XML.

Aby mohl být dokument XML pokládán za správně strukturovaný, musí být dodržena syntaktická pravidla stanovená konsorciem W3C v doporučení XML 1.0. Neoficiálně znamená fráze „správně strukturovaný“ především to, že dokument obsahuje jeden nebo více elementů, přičemž jen jeden z nich musí obsahovat všechny ostatní elementy. Tento element se nazývá kořenový (root element). Kromě toho musí být všechny elementy správně ohraničeny příslušnými značkami – tagy.

Na tvorbu samostatných stránek celého portálu jsou kladeny vysoké požadavky a nároky, neboť by měly být postaveny na XHTML Transitional. Musí tedy respektovat striktní zásady XHTML, které vychází z HTML a XML. Tyto požadavky mají svoje opodstatnění, neboť výsledný formát (XHTML) neobsahuje žádné chyby, které se běžně vyskytují u klasických HTML dokumentů (neukončené tagy, překrývající se vnoření, apod.).

Výsledky

WWW portál Agris je vytvořen pomocí skriptovacího jazyka PHP 4.3 v kombinaci s Microsoft Internet Information Serverem a Microsoft SQL Serverem 2000. Objektový návrh celého portálu je zobrazen na obrázku č. 1, kde největší část služeb celého portálu pokrývají třídy Texts, Links, Weather a Companies.



Obrázek č. 1 – Objektový návrh portálu Agris

Agris je zpravodajským portálem pro agrární sektor, a proto je největším celkem databáze textů a odkazů. Zároveň také disponuje databází firem a aktuálním počasím. Všechny tyto portálové elementy využívají databáze a měly by proto při svém rozsahu umět vyhledávat a klasifikovat obsažené informace. Třídy reprezentující tyto hlavní elementy portálu umožňují vlastní výběry z databáze a jejich prezentaci v podobě XHTML stránek.

Databázi textů je z důvodu technické realizace velmi důležité držet v určité zformátované a dále zformátovatelné podobě. Proto jsou uloženy v XML a lze je tedy zformátovat jako data po jejich jednotlivých elementech.

Agrární www portál Agris využívá definice XML podle News Industry Text Format (NITF), jenž text formátuje vlastními značkami, které se potom při zobrazení textu uživateli portálu transformují do XHTML. XSLT šablon je využíváno právě pro přeměnu XML v XHTML a vlastní převod probíhá na serveru pomocí PHP. Velmi pružné reagování na případné změny v požadavcích na zobrazení textu je nespornou výhodou tohoto systému. Uskutečnění změny stylu zobrazení všech tabulek, obrázků, odstavců, či doplnění určitého textu lze provést s minimálním úsilím, pouze pomocí změny v příslušné XSLT šabloně, přičemž zdrojový text v XML zůstane stejný.

Závěr

Při tvorbě www portálů a jiných rozsáhlejších www aplikací je velmi vhodné využít OOP. Účelně a vhodně se OOP projevil i při realizaci agrárního www portálu Agris, kdy veškeré změny, které se promítají do funkčnosti celého portálu je možné spravovat pouze na jednom místě. Tvorba stránek pro jednotlivé rubriky portálu, popř. rozšíření portálu je pak velice rychlá a efektivní a byla realizována pomocí skriptovacího jazyka PHP 4.3 a databázového serveru MS SQL Serveru 2000.

Zavedení formátu XML pro uchovávání dat v portálu je výrazným krokem vpřed, který může zpřístupňovat texty nejen skrze klasický web, ale i přes Wap, PDA, či jiná zařízení.

Nový koncept se vyznačuje především přívětivějším uživatelským prostředím a také univerzálností a modularitou programového řešení.

Použitá literatura

1. POLÁK, J.; MERUNKA, V.; CARDA, A. Umění systémového návrhu – Objektově orientovaná tvorba informačních systémů pomocí původní metody BORM. Praha : Grada Publishing a. s., 2003. 196 s. ISBN 80-247-0424-2.
2. CASTEGNETO, J.; RAWAT, H.; SCHUMANN, S.; SCOLLO, C.; DEEPAK, V. Programujeme PHP profesionálně. Praha : Computer Press, 2001. 656 s. ISBN 80-7226-310-2.
3. HOLZNER, S. XSLT – Příručka internetového vývojáře. Praha : Computer Press, 2002. 516 s. ISBN 80-7226-600-4.
4. Internet Scout Project, Department of Computer Sciences, University of Wisconsin-Madison [online]. [cit. 15. 12. 2002] <<http://scout.cs.wisc.edu>>.
5. LAGOZE, C.; SOMPEL, H.; NELSON M.; WARNER, S. The Open Archives Initiative Protocol for Metadata Harvesting [online]. posl. revize 14. 6. 2002 [cit. 14. 12. 2002] <<http://www.openarchives.org/OAI/2.0/openarchivesprotocol.htm>>.
6. CORMEDIA - Document and Content Management Solutions - Internet portals [online]. [cit. 20. 12. 2002] <<http://www.cormedia.com.sg/DOCPortal.htm>>.
7. TRAFFICK - The Guide to Portals & Search Engines [online]. [cit. 2. 1. 2003] <<http://www.traffick.com/directory/portals/default.asp>>
8. GAUSS - Issues to consider when defining your Enterprise Portal strategy [online]. [cit. 16. 12. 2002]
9. PETERKA, J. Od distribuce informací po podnikové portály. In Softwarové noviny. č. 5 (květen), roč. 13, s. 10-15. ISSN 1210-8472.
10. HOMAN, D.; SANCHEZ, E.; KLIMA C.; Building A Portal? Vive La Différence [online]. Publ. 1. 11. 2001, posl. revize: 30. 1. 2002 [cit. 21. 12. 2002] <<http://www.informationweek.com/story/IWK20011101S0014>>.