

1

**Vysoká škola ekonomická v Praze
Fakulta informatiky a statistiky
Vyšší odborná škola informačních služeb v Praze**

Jan Hromádko

**Tvorba portálu sdílených znalostí pro
průmyslový klastr OMNIPACK**

Bakalářská práce

2006

Prohlášení

Prohlašuji, že jsem bakalářskou práci na téma „Tvorba portálu sdílených znalostí pro průmyslový klastr OMNIPACK“ zpracoval samostatně a použil pouze zdroje, které cituji a uvádím v seznamu použité literatury.

V Praze, dne 28. května 2006

Podpis

Poděkování

Za cenné připomínky a rady při zpracování bakalářské práce bych rád poděkoval vedoucí bakalářské práce paní PhDr. Kateřině Julišové.

Obsah

1	Úvod- o čem je moje práce	6
2	Teoretická rovina projektu	7
2.1	Průmyslový klastr Omnipack.....	7
2.2	Topic Maps (mapy námětů).....	8
2.3	Znalostní portál	9
3	Limity a omezení projektu.....	11
4	Fáze vývoje portálu.....	11
4.1	První fáze	11
4.1.1	Databáze.....	14
4.1.2	Tabulky	15
5	Druhá fáze.....	17
6	Třetí fáze	17
7	Čtvrtá fáze	18
8	Pátá fáze.....	20
9	Šestá fáze	20
10	Sedmá fáze	20
11	Používané příkazy SELECT, UPDATE a DELETE	21
12	Technický popis enginu portálu	23
13	Grafický layout.....	26
14	Shrnutí vývoje portálu	30
15	Znalostní portál a jeho komerční nasazení	31
16	Závěr	33
17	Seznam použitých zdrojů	35
18	Přílohy.....	37

1 Úvod- o čem je moje práce

Má práce popisuje vývoj a řešení *Znalostního portálu pro potřeby průmyslového klastru OMNIPACK*, který jsem vytvořil během své studijní praxe ve firmě Tovek s.r.o. v souvislosti s grantem AV ČR „*Znalostní báze pro vyhledávání a rozvoj synergických podnikatelských seskupení*“. Cílem mé práce bylo pokusit se využít některé principy popsané v rámci zmíněného grantu při tvorbě znalostního portálu, a tím se de facto pokusit ověřit, zda jsou aplikovatelné v praxi. Jedná se především o principy spojené se standardem Map námětů (*topic maps*) a dále o principy, o které by se měl opírat *znalostní portál*. V praktické rovině se potom snažím při využití získaných poznatků a informací vytvořit funkční prototyp zmíněného znalostního portálu.

Celý projekt měl poměrně komplexní charakter a při jeho řešení jsem se potýkal s mnoha problémy od čistě teoretických až po ty praktické. Největší překážkou se ukázala být samotná definice toho, co má být výsledkem mé práce. Jelikož jsem pracoval na grantu, který byl stále ve své první fázi, a nebyl tím pádem k dispozici žádný oficiální výstup, byl jsem odkázán pouze na osobní konzultace s jednotlivými řešiteli a na některé již dokončené (byť neoficiální) výstupy, zejména pak na ukázkový model portálu, jehož tématickou strukturu jsem převzal.

Hlavní výzvou pro mne bylo pokusit se porozumět značně rozdílným a často poměrně obecným představám jednotlivých řešitelů a na základě toho vytvořit konkrétní a alespoň částečně funkční prototyp portálu, který by bylo možno využít jako jeden ze základů při snaze o praktické budoucí využití principů v grantu popsaných. Jak jsem toto vše řešil a zda jsem při tom obstál, je hlavní náplní praktické roviny mé bakalářské práce.

2 Teoretická rovina projektu

2.1 Průmyslový klastr Omnipack

Na stránkách agentury Czechinvest, která je v Čechách hlavním propagátorem a organizátorem klastrových iniciativ, se dočteme, že ¹: „*Klastr je soubor regionálně propojených společností (podnikatelů) a přidružených institucí a organizací – zejména institucí terciárního vzdělávání (VŠ, VŠO) - jejichž vazby mají potenciál k upevnění a zvýšení jejich konkurenceschopnosti.*“

Klastry jsou v Čechách zatím stále jen v počáteční fázi rozvoje, s výjimkou právě klastru Omnipack, který svojí počáteční fází prošel a obdržel již i dotaze EU na svůj rozvoj. Kalstr Omnipack sdružuje 22 subjektů jak z podnikatelské, tak z akademické a státní sféry a mezi jeho partnery patří kromě agentury Czechinvest např. Fakulta ekonomicko-správní Univerzity Pardubice, dále Fakulta strojní a Katedra průmyslového inženýrství a managementu Západočeské Univerzity v Plzni, Rada Královéhradeckého kraje, Ministerstvo průmyslu a obchodu a další.

Podniky sdružené v tomto klastru si sice vzájemně konkurují, zároveň však pro ně klastrové uskupení představuje jedinečnou šanci, jak překonat řadu nevýhod, které jim jejich malá či střední velikost oproti velkým podnikům přináší, a posílit tak svoje šance na úspěch v podmínkách globální ekonomické soutěže. Jak konkrétně tato spolupráce zatím vypadá? Zúčastněné podniky například spolufinancují společnou prezentaci na mezinárodních veletrzích, na kterou by jednotlivě neměly dostatek prostředků. Ještě důležitějším bodem spolupráce je potom výstavba společného vývojového a zkušebního centra. To poskytne členům klastru špičkově vybavené zázemí pro rozvoj nových technologií, a to i ve spolupráci s akademickou sférou. Mezi další aktivity patří Marketingový a obchodní portál klastru, vyhledávání a výzkum inovací ve spolupráci s univerzitou Pardubice, vydávání společných propagačních materiálů a podobně.

Význam klastrů bude do budoucna rozhodně stoupat. Stále se přiostrhující celosvětový konkurenční boj ničí malé hráče, kteří nemají moc možností, jak se tomuto vývoji bránit. Jednou z možností je fúze s větším podnikem, podobné

¹ Czechinvest, Klastry

řešení však není pro každého. Další možností je tedy spolupráce v rámci klastru. Společná spolupráce na vývoji nových technologií a na propagaci vlastní produkce může přinést velmi pozitivní výsledky. Když navíc takto sdružené podniky začnou využívat nejmodernější technologie pro sdílení a rozvoj znalostí, může jim to přinést společnou výhodu, která pro ně bude znamenat budoucnost a prosperitu, místo postupného rozpuštění v globálním ekonomickém moři. Vždyť znalosti bývají označovány ² jako čtvrtý výrobní faktor a právě jejich účinné shromažďování a využití bude v budoucnu pro mnohá odvětví klíčovou záležitostí.

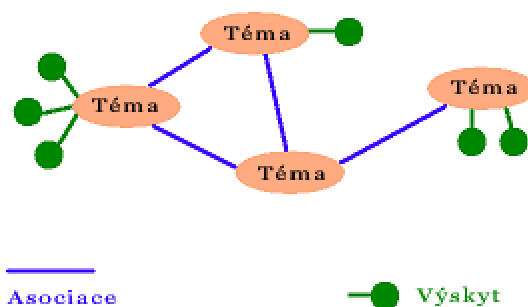
2.2 Topic Maps (mapy námětů)

Nejdříve k principům a vlastnostem Topic maps podle standardu ISO/IEC 13250:2003- *“Sada jednoho nebo více vzájemně souvisejících dokumentů, která používá zápis definovaný tímto mezinárodním standardem, se nazývá mapa námětů“* ³ a dále- *„Obecně informace o struktuře uložená v mapě námětů obsahuje seskupení adresovatelných informačních objektů kolem témat (výskyty) a vztahy mezi těmito tématy (asociace).“* Dle definice *„Mapa námětů popisuje prostor námětů o mnoha rozměrech- prostor, v němž místa jsou náměty a ve kterém je vzdálenost mezi náměty měřitelná počtem intervenujících námětů, které musí být navštíveny při cestě od jednoho námětu k druhému, a druhy vazeb, které určují cestu od jednoho námětu k druhému“*, přičemž na významu nic nezmění, použijeme-li místo slova *námět* slovo *téma*.

Ve srozumitelnější formě vyjadřují uvedené definice zhruba toto: Základním prvkem mapy námětů (topic map) je *téma* neboli předmět, kterému bývá přiřazen konkrétní informační zdroj (potom se jedná o výskyt). Témata jsou propojena asociacemi, které mohou mít rozdílné vlastnosti (např. vztah předek/potomek). Témata a vztahy mezi nimi tvoří znalost, výskyty tuto strukturu propojují s informační rovinou.

² T. C. A. Ramanujam, Into the fourth dimension of economics

³ IEC 13250:2003



Obr. 1 Mapy námětů (převzato z http://en.wikipedia.org/wiki/Topic_maps)

Právě tyto principy jsem při vytváření architektury portálu využil, byť v poněkud zjednodušené formě. Omezil jsem se totiž na pouhé dva elementy – téma a asociaci. Každá asociace přitom uchovává pouze informaci, jaká dvě témata propojuje, a tím pádem také informaci o hierarchické struktuře témat, která je však využita jen v omezené míře (viz dále). Téma pak obsahuje svůj vlastní popis a dále i případný odkaz na externí výskyt. Konstrukce portálu je takto maximálně odlehčena a zjednodušena, zachovává si však základní vlastnost – vytváří přehlednou strukturu, která se může libovolně průběžně modifikovat, doplňovat a propojovat podle potřeb uživatelů.

2.3 Znalostní portál

Co to vlastně je znalostní portál a jak by měl vypadat? Znalost je pochopená informace zařazená do širších souvislostí. Znalostní portál by tedy měl být vytvořen tak, aby informace v něm obsažené získávaly strukturou a způsobem své prezentace přidanou hodnotu. Uživatel by při hledání konkrétní informace měl zároveň s jejím nalezením získat i přehled o souvislostech a vazbách s touto informací spojených. Jak by však tato teorie měla vypadat v praxi? Základním předpokladem je logická, jednoduchá, přehledná a zároveň vyčerpávající struktura jednotlivých tematických kategorií. Tato struktura, vytvořená řešiteli grantu ve spolupráci s odborníky z prostředí klastu Omnipack, mi byla k dispozici a posloužila bez jakékoliv změny k vytvoření kostry portálu.

Čím však tuto kostru obalit, jakým způsobem dát portálu obsah? Ve svém řešení jsem se spolehl pouze na interní zdroj informací – tedy pouze na informace, které do portálu vloží samotní uživatelé, případně správci portálu. Toto nedostatečné řešení jsem zvolil zejména z časových důvodů a jeho neúplnosti jsem si byl od začátku vědom. Externí zdroje by totiž měly tvořit nedílnou součást každého znalostního portálu. Mají-li být informace předkládány co nejkomplexněji a v co největší šíři, nelze se spolehnout jen na jejich interní zdroje, ale do prostředí portálu musí být integrována co nejpestřejší paleta externích databází a dalších informačních zdrojů. Jeden příklad za všechny – databáze filmových titulků opensubtitles.org obsahuje kromě samotných titulků i vyčerpávající řadu dalších informací z externích databází, které jsou vynikajícím způsobem integrovány na jediné zobrazené dynamicky generované stránce. Nejen že si tedy uživatel může snadno vyhledat požadované titulky, ale zároveň vidí všechny důležité údaje o filmu a má navíc k dispozici celou řadu odkazů vedoucích k vyčerpávajícím informacím.

Dále je potřeba odhadnout, komu bude takový portál vlastně sloužit – zřejmě především majitelům, manažerům, ale i zaměstnancům firem v rámci klastru. Jak zajistit, aby se tito uživatelé, pro které byl portál vytvořen a kteří budou z části jen základně počítačově gramotní, naučili využívat jeho možností a časem se dokonce aktivně a na dobrovolné bázi podíleli na jeho obsahu? Důležité je samozřejmě splnit obecné nároky, které jsou dnes na moderní webová řešení kladeny – přehlednost, přitažlivý grafický design, uživatelskou jednoduchost, rychlost, spolehlivost atd. Nutná by byla jistě možnost personalizace uživatelského rozhraní jak po funkční, tak po grafické stránce a také funkce systému, které by uživatele nutily se do něj pravidelně vracet – především interní pošta a dále pak např. zajištění kanálu s aktuálními informacemi z oboru, za který by si jinak museli uživatelé sami platit. Předchozí text výstižně shrnuje následující definice 4 :

Podnikový (tato definice se jistě nezmění ani pro portál klastrový) znalostní portál by se měl skládat z těchto částí:

⁴ NSS group white paper

Informačního podnikového portálu, který nabízí přihlášeným uživatelům na základě jejich dotazů personalizované informace.

Portálu pro podnikovou spolupráci, který lidem nabízí virtuální prostory ke spolupráci.

Odborného podnikového portálu, který propojuje lidi na základě jejich schopností.

Podnikový znalostní portál nabízí vše zmíněné a navíc v reálném čase přináší propojení na obsah a osoby, které jsou relevantní právě ve vztahu k úlohám, které uživatel řeší.

3 Limity a omezení projektu

Hlavní omezení pro mě znamenaly následující faktory – čas (pouze cca. 6 týdnů), omezená zkušenost s používanými technologiemi a nejasný cíl práce. Kvůli omezenému časovému rámci bylo nutné jednotlivé prvky zjednodušit a redukovat na jejich samotnou podstatu. Nebylo například možno ošetřit všechny možné programové výjimky (což je za normálních okolností samozřejmě naprosto nezbytné), vytvářet obsáhlý komentář ke kódu, vymýšlet propracovaný grafický layout, optimalizovat databázi, či zajišťovat validitu HTML kódu. Jedině tak se podařilo v daném termínu vytvořit výsledek, který bylo možno prezentovat ve funkční podobě.

4 Fáze vývoje portálu

4.1 První fáze

Nejdříve bylo nutné zjistit, co má vlastně být cílem mé práce. V tomto jsem byl odkázán na členy grantového týmu, konzultace s nimi však až na jedinou výjimku mohly kvůli jejich pracovní zaneprázdněnosti probíhat pouze sporadicky. Zvolil jsem kvůli tomu postup, při kterém jsem všechny části portálu implementoval co nejobecněji a nejuniverzálněji. Chtěl jsem se tak vyhnout situaci, kdy bych se po

několika týdnech vývoje dozvěděl, že postupuji špatně – takové zdržení by se již dalo dohnat těžko. Zvolený postup se osvědčil a dodatečné požadavky na portál se týkaly spíše doplnění funkcí, nežli jejich výrazné úpravy, úplné změny nebo dokonce odstranění.

Postup byl tedy alespoň částečně jasný, ne však cíl a očekávaný výsledek, jak jsem záhy zjistil. Představy řešitelů grantu, s nimiž jsem projekt portálu konzultoval, byly dosti rozdílné a navíc nekonkrétní, což je ale u nově vyvíjeného řešení celkem pochopitelná věc. Jediné z čeho se dalo vycházet, bylo následující schéma, které bylo již pevně dané:

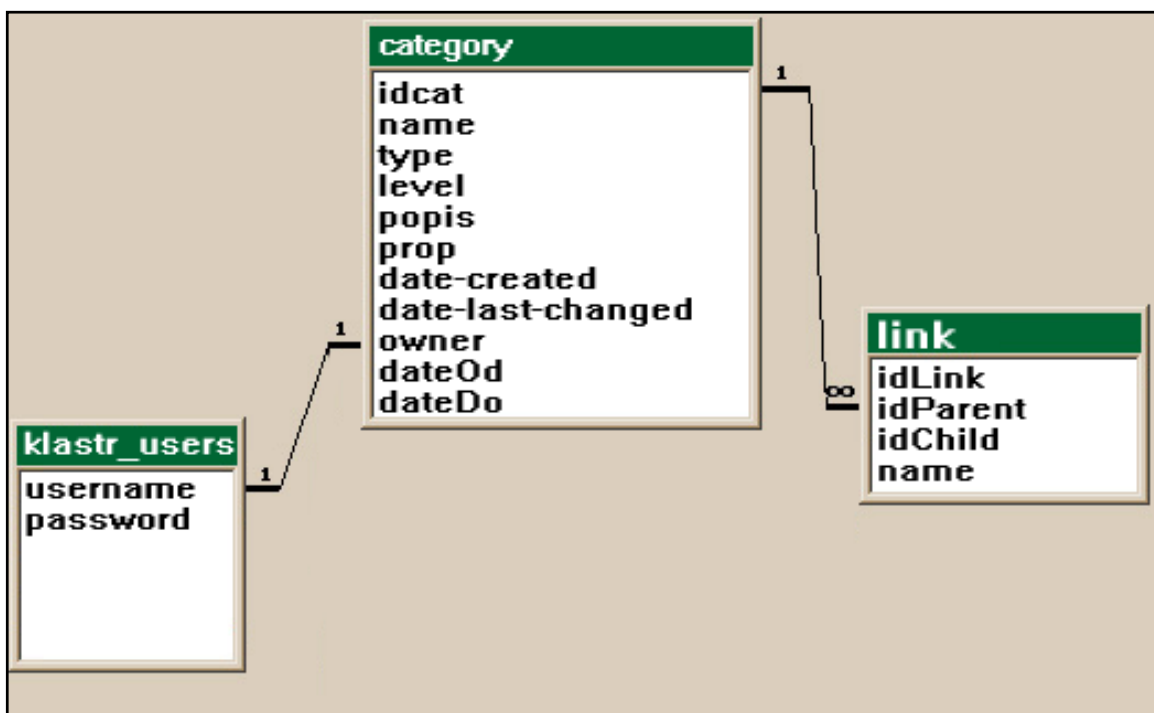
Toto schéma vytvořili řešitelé grantu ve spolupráci s odborníky z řad klastru Omnipack. Jeho vývoji věnovali značné úsilí tak, aby se jim podařilo co nejlépe zachytit vztahy a vazby, které jak v rámci klastru, tak mezi klastrem a jeho okolím existují. Je zřejmé, že základem znalostního portálu se může stát jediné struktura, v níž už bude nějaká znalost uložena. V tomto jediném bodě jsem se proto vyvaroval jakýchkoliv vlastních zásahů a plně jsem se podřídil zvolenému schématu, neboť o problematice jsem neměl dostatečné informace a jakýkoliv zásah by tedy byl nejspíše ke škodě věci, byť jsem se ze začátku o drobné změny pokoušel. Byly však odmítnuty a záhy jsem pochopil, že schéma, jež se mi na první pohled nezdálo přesvědčivé, je vytvořeno skutečně důmyslně tak, aby odpovídalo co nejvíce potřebám praxe.

Výše zobrazené schéma bylo vytvořeno pomocí špičkového analytického nástroje Analyst's notebook od firmy i2. Tento nástroj je sice primárně určen pro analýzy poněkud odlišného typu (především z oblasti zpravodajství), ovšem i nasazení v tomto případě bylo opodstatněné, jak bude vysvětleno později. Analyst's notebook je vizuální nástroj pro analýzu souvislostí. V tomto případě byl pomínut časový rozměr analyzovaných témat a typy vkládaných entit byly omezeny na sedm – osoba, místo, organizace, věc, proces, událost, kategorie. Pro převod schématu do databáze jsem pak zvolil pouze entity typu kategorie, ostatní měly spíše pomocný pracovní charakter a jejich převod nebyl podstatný. Aby bylo možné již připravená data někam převést, bylo dále nutné navrhnout a vytvořit vhodnou databázi.

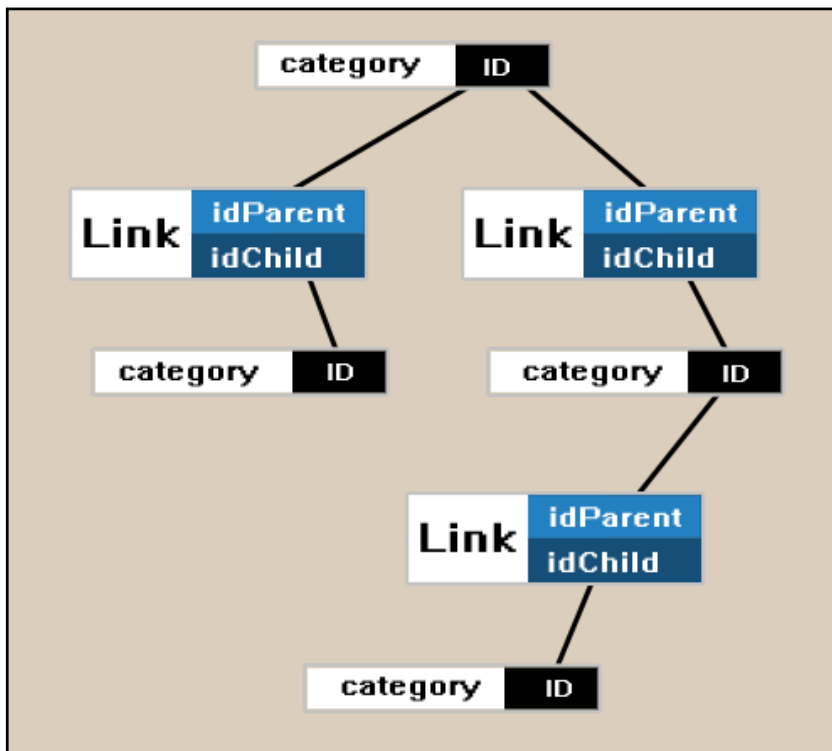
4.1.1 Databáze

Databázi jsem vytvořil na platformě MySQL a to zejména proto, že tato technologie je k dispozici zdarma a měl jsem s ní navíc zkušenosti již z dřívější doby.

Kompletní databázi tvoří pouze tři tabulky – dvě základní, ve kterých je uložena struktura a obsah portálu, a jedna pomocná pro uložení informací o uživateli a jejich účtech.



Každá entita (z tabulky category) musí mít právě jednoho vlastníka, který odpovídá některému z uživatelů z tabulky KALSTR_USERS. Každá entita také musí mít minimálně jednoho rodiče. Engine portálu totiž pracuje tak, že při zobrazování prochází postupně strukturou entit a jejich spojení, takže entita bez propojení s nějakým rodičem je neviditelná a nebude nikdy zobrazena. Potomků může mít každá entita libovolný počet, ale nemusí mít žádného.



Obr. 4 Princip propojení entit

4.1.2 Tabulky

Tabulka KLASTR_USERS uchovává pouze dvě informace – jméno uživatele a přidělené heslo. Primární klíč je na sloupci USERNAME. Toto je opravdu to nejjednodušší řešení, pro praktické použití by mělo být o uživateli uchováno mnohem více informací – datum a IP adresa posledního přihlášení, poskytnuté osobní a kontaktní informace a uživatelovo osobní nastavení. Heslo by také mělo být uloženo v šifrované podobě.

Field	Type	Collation	Attributes	Null	Default	Extra
<u>username</u>	varchar(30)	cp1250_general_ci		No		
password	varchar(30)	cp1250_general_ci		No		

Tabulka CATEGORY s primárním klíčem na sloupci IDCAT (s volbou auto increment) obsahuje informace o všech entitách – jméno, typ (1-7), dále pak úroveň v navigaci, vlastní popis entity, pro budoucí použití vlastnosti, datum vytvoření, datum poslední úpravy, vlastník (ten kdo entitu vytvořil). Sloupce dateOd a dateDo slouží k uchování časových údajů, vztahuje-li se entita k nějakému datu nebo časovému intervalu.

Field	Type	Collation	Attributes	Null	Default	Extra
idcat	smallint(6)			No		auto_increment
name	varchar(100)	cp1250_general_ci		No		
type	tinyint(15)			No		
level	smallint(6)			No		
popis	text	cp1250_general_ci		Yes	NULL	
prop	text	cp1250_general_ci		Yes	NULL	
date_created	datetime			No		
date_last_changed	datetime			No		
owner	varchar(20)	cp1250_general_ci		No		
dateOd	date			Yes	NULL	
dateDo	date			Yes	NULL	

Tabulka LINK obsahuje údaje o propojení všech entit. Primární klíč je na sloupci idLink (opět s volbou auto increment). Sloupec NAME by měl v budoucnu obsahovat typ vazby, podle kterého se určí role jednotlivých entit.

Field	Type	Collation	Attributes	Null	Default	Extra
idLink	smallint(6)			No		auto_increment
idParent	smallint(6)			No		
idChild	smallint(6)			No		
name	varchar(100)	cp1250_general_ci		Yes	NULL	

Vzhledem k malému objemu mnou používaných testovacích dat (tabulky LINK a CATEGORY kolem 300 záznamů, KLASTR_USERS 3 záznamy) nebylo nutné nijak řešit optimalizaci, ani vytvářet další indexy. Ovšem ani při ostrém nasazení tohoto řešení by zřejmě nebyl s rychlostí problém, protože počet záznamů by byl stále jen v řádu tisíců, použité příkazy typu SELECT jsou poměrně jednoduché a prováděné kartézské součiny se týkají maximálně dvou tabulek současně.

5 Druhá fáze

První fáze tedy byla dokončena – měl jsem k dispozici výchozí strukturu a vlastní databázi (zatím prázdnou). Ani po dlouhých debatách jsme se ovšem s kolegy nebyli schopni shodnout, co je vlastně podstatou znalostního portálu. V tomto ohledu nebylo moc platné ani hledání na internetu – názorů je celá řada a jako znalostní portál bývá označováno mnoho rozdílných věcí. O tom proč to tak je, se zmiňuji již dříve – znalostní portál zkrátka netvoří jedna konkrétní vlastnost nebo funkcionalita, ale celý komplex funkcí a vlastností, který když je správně nastaven, může být označen slovem znalostní. Na druhou stranu se věci nemají tak, že něco by na 100% buď bylo nebo nebylo „znalostní“. Systémy se vždy nacházejí na pomyslné cestě mezi informacemi a znalostmi. Moje znalosti byly malé, a proto jsem jich do systému mnoho vložit nemohl. Mohl jsem však vytvořit prostředí, ve kterém budou toto dělat sami uživatelé, kteří se z podstaty věci v problematice více orientují. Toto je velmi podstatná myšlenka – je nutné vědět, do čeho se má cenu pouštět a do čeho ne, a to teď již bylo jasné. Těžko lze přinést nové znalosti v oblasti, ve které nemám orientaci a se kterou nemám zkušenosti, navíc během několika týdnů. Lze ovšem zařídit, aby toto byli schopni udělat k tomu povolání uživatelé.

S ostatními řešiteli jsem se tedy dohodl na následujícím postupu. Již dříve byla vytvořena jakási „maketa“ portálu. Jednalo se o statické stránky, ovšem vygenerované na základě schématu uloženého v Analalyst's notebooku. Mým úkolem bylo teď vytvořit jejich obdobu, nyní ovšem plně dynamickou, s možností přímé editace ze strany uživatelů a v odlišné grafické podobě. Zprvu značně nejasný úkol tedy konečně nabyl konkrétní podoby a mohl jsem se pustit do další etapy.

6 Třetí fáze

Od primárního zadání uběhl zhruba týden a já měl k dispozici strukturu portálu, databázi a nyní již poměrně konkrétní cíl, což by se dalo za daných podmínek považovat za slušný krok kupředu. Hlavní díl práce byl ovšem ještě celý přede

mnou – naprogramovat engine portálu, tedy program, který bude podle obsahu databáze portál graficky generovat a umožní uživateli obsah databáze měnit.

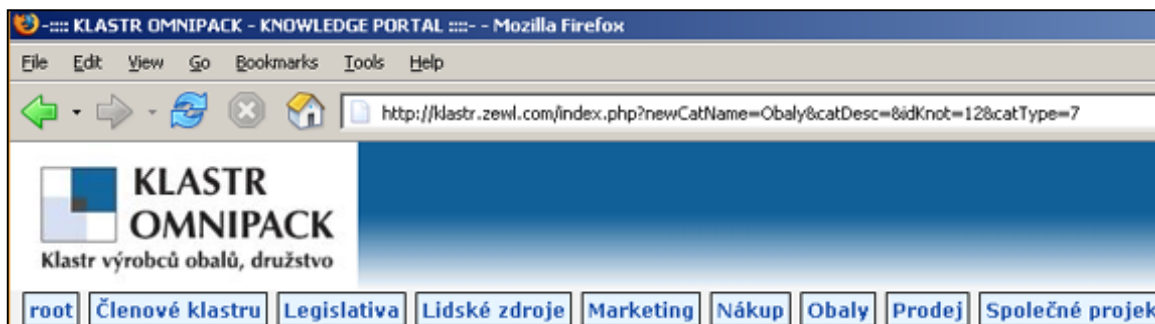
Rozhodl jsem se pro použití jazyka PHP, samozřejmě v kombinaci s HTML a CSS. Opět zejména proto, že jsem s tím měl z dřívějšíka největší zkušenosti. Protože jsem byl na práci sám a jen těžko se dalo odhadnout, kam až se mi ji podaří dotáhnout, zvolil jsem následující metodu softwarového vývoje, která by se dala popsat přirovnáním ke sněhové kouli – vytvořit úplně jednoduchý základ engine, na který budu postupně „nabalovat“ další a další funkce, dokud budu mít dostatek času, to vše samozřejmě při pokud možno co nejčastějších konzultacích s řešiteli grantu. Tento postup měl několik zásadních výhod – portál byl v krátké době už alespoň částečně funkční a od té doby jsem měl jistotu, že ať už bych skončil s vývojem kdykoliv, budu schopný odevzdat a předvést v provozuschopné podobě alespoň něco. Zadavatelé byli také schopni od začátku sledovat vývoj práce a náležitě ho usměrňovat, navíc měli slušný přehled o jejím dosavadním rozsahu, a protože jsem na toto téma z jejich strany nezaznamenal negativní komentář, nemusel jsem se obávat, že moje práce bude při předání hodnocena jako nedostatečná.

7 Čtvrtá fáze

V předešlé fázi jsem volil obecnou metodiku, nyní bylo potřeba přejít ke konkrétnějším krokům. Původně jsem zamýšlel následující postup – nejdříve ručně převést převzaté schéma kategorií do vlastní databáze, na těchto datech pak zkoušet různé varianty zobrazení a nakonec přidat možnost editace těchto dat. Ruční plnění databáze pomocí rozhraní phpMyAdmin (verze 2.9.1.1), které jsem se chystal pro tento účel využít, by bylo zbytečně zdlouhavé, a tak jsem se rozhodl postup změnit – nejdříve naprogramovat rozhraní pro vkládání dat, potom zajistit jejich zobrazení a nakonec možnost jejich editace.

Analyst's notebook (ve kterém bylo převáděné schéma vytvořeno) ukládá svoje data do databáze iBase od stejného výrobce a neumožňuje jejich použitelný export do jiné databáze. Bohužel ani export dat z iBase v nějakém standardním formátu není možný, zde mi tedy pomohl kolega. Ten měl z minulosti hotový skript ve VisualBasicu, který postupně četl údaje ze speciálního textového souboru

exportovaného z iBase a přes internetový prohlížeč je postupným zadáváním modifikovaných údajů do adresové řádky vkládal přímo do databáze, tedy již za spolupráce s engineem portálu.



Tato metoda zkrátila převod dat z hodin na několik minut (počítaje v to i nutnou úpravu VBasic skriptu). Navíc je při použití zmíněných technologií univerzální a lze jí použít pro vytvoření libovolné struktury. Bohužel zde není zajištěna možnost exportu dat druhým směrem, tedy aby se změny provedené na portálu promítly do struktury v iBase, implementovat tuto funkčnost by bylo kvůli povaze záznamů iBase značně obtížné.

Členové klastru	Jádro klastru	Okolí klastru
Členové klastru	Jádro klastru	Okolí klastru
Jádro klastru	Členové klastru	
Jádro klastru	Členové klastru	
Okolí klastru	Členové klastru	
Okolí klastru	Členové klastru	
Společné projekty	Marketingové a obchodní centrum	Logistické a ně
Společné projekty	Marketingové a obchodní centrum	Logistické a ně
Marketingové a obchodní centrum	Tuzemské veletrhy a výstavy	Zahrani
Marketingové a obchodní centrum	Tuzemské veletrhy a výstavy	Zahrani
Tuzemské veletrhy a výstavy	Marketingové a obchodní centrum	
Tuzemské veletrhy a výstavy	Marketingové a obchodní centrum	
Zahraniční veletrhy a výstavy	Marketingové a obchodní centrum	
Zahraniční veletrhy a výstavy	Marketingové a obchodní centrum	
Marketingové informace o trzích podle priorit	Rusko, Srbsko-Černá Hora	
Marketingové informace o trzích podle priorit	Rusko, Srbsko-Černá Hora	
Rusko, Srbsko-Černá Hora, Francie, Benelux, Španělsko, GB, Irsko		
Rusko, Srbsko-Černá Hora, Francie, Benelux, Španělsko, GB, Irsko		

Obr. 9 Ukázka dat iBase

8 Pátá fáze

Databáze je již naplněna relevantními daty a je zajištěno i jejich vkládání, nyní bylo nutné zajistit jejich adekvátní zobrazení. Mým cílem přitom bylo zajistit maximální univerzálnost a flexibilitu řešení. Vytvořil jsem tedy sadu funkcí, které vždy generovaly v závislosti na obsahu databáze konkrétní část portálu (navigace, kořenové kategorie). Potom jsem vytvořil hlavní stránku, jejímž základem je poměrně jednoduchá HTML tabulka, obsahem jejích polí je pak pouze volání vybrané funkce (nebo více funkcí) a případně nadpis, jenž ukazuje na povahu vypisovaných dat. Zobrazená stránka tedy obsahuje minimum textu, který by nebyl vygenerován v souvislosti s obsahem databáze. Pořadí, umístění a popis jednotlivých prvků je navíc velmi snadné změnit, stačí k tomu pouhý přesun volání funkce do jiného pole tabulky, nebo obyčejná změna textu popisu v HTML kódu. Řešení je tak opravdu flexibilní, co se obsahu zobrazovaných dat týče (pokud jsou stejné povahy jako v řešeném případě), a cíl 5. fáze byl tedy splněn.

9 Šestá fáze

Data jsou uložena a podle volby uživatele i zobrazena, zbývá zajistit možnost jejich editace. I zde jsem použil maximálně jednoduchý způsob. Zvolí-li uživatel možnost úpravy aktuálně zobrazené entity, všechny údaje, jež má právo měnit, jsou vypsány do formuláře, kde je možnost je upravit a následné změny potom uložit. Vytváření i mazání entit probíhá na podobném principu. Dále jsem přidal možnost vytváření nebo rušení vazeb mezi entitami. Zde je nutné, aby byl zvolen mód linkování, ve kterém nejdou jiné úpravy provádět.

10 Sedmá fáze

Engine portálu byl již v tuto chvíli plně funkční, rozhodl jsem se ale ještě doplnit několik funkcí, které zvýší uživatelský komfort a usnadní navigaci. Přibyly možnost vypsání místo stromové navigace abecedně celý obsah databáze a tlačítka, která najednou buďto rozbalí nebo sbalí celou navigaci. Přidal jsem také tlačítka se

symboly entit, která mají funkci filtru, který z databáze abecedně vypíše entity zvoleného typu. Tím bylo uživatelské rozhraní z hlediska obsahu hotovo, v následující části se věnuji podrobněji technické stránce věci.

11 Používané příkazy SELECT, UPDATE a DELETE

Dotaz, který vyfiltruje všechny zadaného typu, abecedně seřazené – jedná se o jednu z možností zobrazení navigace, typ proměnné je předán pomocí proměnné \$sw.

```
SELECT category.name, category.idcat FROM category WHERE  
(category.type='$sw') ORDER BY name asc
```

Dotaz pro zobrazení části navigace – vypisuje všechny potomky aktuálního uzlu, které jsou typu 7 – kategorie, tedy jen takové, které tvoří navigaci a řadí je abecedně.

```
SELECT category.name, category.idcat, category.level,  
category.type FROM category, link WHERE (link.idChild =  
category.idcat AND link.idParent = '$idKnot' AND category.type=7)  
ORDER BY name ASC
```

Dotaz směřující ke zjištění počtu potomků typu kategorie (7) stejného typu. Slouží při zobrazování navigace – nemá-li kategorie potomky, je v navigaci obvyčejně vypsána. Má-li nějaké potomky, je před její název zařazen grafický symbol + nebo - , podle toho, jak je navigační strom rozbalen.

```
SELECT COUNT(*) FROM (category JOIN link ON category.idcat =  
link.idChild) WHERE (link.idParent = '$rs[idcat]' AND  
category.type=7)
```

Tento dotaz vypíše pro účely navigace všechny potomky aktuální entity, která je dotazu předána v proměnné \$idKnot.

```
SELECT category.name, category.idcat, category.type FROM category,  
link WHERE (link.idParent = '$idKnot' AND  
category.idcat=link.idChild)
```

Tento dotaz vrátí všechny záznamy, jejichž položka dateOd je menší než aktuální datum (a tedy už „začaly“), a naopak položka dateDo je větší než aktuální datum (a zaznamenaná entita tedy právě „probíhá“).

```
SELECT category.name, category.idcat, category.type, dateOd,
dateDo FROM category WHERE (CURDATE() >= dateOd AND CURDATE() <=
dateDo) ORDER by dateDo ASC
```

Stejně jako předchozí dotaz zjišťuje vztah entity k aktuálnímu datu – když je aktuální datum menší, znamená to, že entita ještě „neproběhla“ a je vypsána.

```
SELECT category.name, category.idcat, category.type, dateOd FROM
category WHERE CURDATE() <= dateOd ORDER BY dateOd ASC
```

Kartézský součin tabulek CATEGORY a LINK, ze kterého se zjistí počet potomků zadané entity ($\$rs[idcat]$). To je důležité, chce-li uživatel nějakou entitu smazat – tuto volbu dostane k dispozici jedině, když nemá entita žádné potomky (když je výsledek dotazu 0).

```
SELECT COUNT(*) FROM category JOIN link ON category.idcat =
link.idChild WHERE link.idParent = '$rs[idcat]'
```

Tento dotaz vypíše všechny potomky entity ROOT, což slouží pro usnadnění navigace.

```
SELECT category.name, category.idcat, category.level,
category.type FROM category, link WHERE (link.idChild =
category.idcat AND link.idParent = 1 AND category.type=7) ORDER BY
type DESC,name ASC
```

Tento dotaz se ptá na předka aktuální entity ($\$predchudce$), rekurzí tohoto dotazu se lze postupně dostat z jakéhokoliv místa portálu až k entitě root.

```
SELECT category.name, link.idParent FROM category, link WHERE
(link.idChild='$predchudce' AND category.idcat='$predchudce')
```

Jednoduchý dotaz, který je použit pouze při vkládání nové entity – jelikož je přidělování unikátního ID v tomto případě řešeno automaticky pomocí AUTO INCREMENT, do přidání další entity má ID té naposledy přidané maximální hodnotu, kterou lze takto snadno zjistit.

```
SELECT MAX(idcat) AS last_id FROM category
```

Základní dotaz, v jednom z módů navigace vypíše abecedně tabulku CATEGORY

```
SELECT type, name, idcat FROM category ORDER BY name ASC
```

Dotaz při ověřování hesla, zabezpečení je v tomto případě opravdu minimální.

```
SELECT password FROM klastr users WHERE username = '$username'
```

12 Technický popis enginu portálu

`index.php` - vypisuje html kód se strukturou tabulky, pro vygenrování jejich obsahu potom volá speciální funkce.

`functions.php` - obsahuje volané funkce.

`database.php` - obsahuje pouze údaje nutné pro připojení k MySQL databázi.

`login.php` - zajišťuje správu uživatelů a jejich hesel, s několika úpravami jsem použil skript, který je volně k dispozici na serveru www.evolt.org⁵

Jak tedy engine portálu pracuje? Základním prvkem je jednoduchá, strukturou stále stejná HTML tabulka, do jejíchž polí se dynamicky generují navigační prvky a obsah databáze podle zadání uživatele. Nejdříve je ale nutné připojit následující soubory:

```
include("database.php");
include("functions.php");
session_start();
include("login.php");
```

Výsledek výše zmíněné autentifikace uživatele je aplikován pomocí jednoduchého větvení:

```
if ($logged_in)
```

Je-li uživatel zalogován, vykoná se tělo programu, když není, jde se přímo na konec skriptu a nevykoná se žádný další příkaz.

Dále je nutné načíst předávané proměnné, podle jejichž obsahu (či absence obsahu) se řídí další vykonávání programu. Tyto proměnné jsou předávány pomocí formulářů a metody POST.

⁵ JP, PHP login script with remember me feature

```

$idKnot = $_REQUEST["idKnot"];
$newCatName = $_REQUEST["newCatName"];
$path_root = '<a href="index.php?idKnot=1">root</a>';
$deleteCat = $_REQUEST["deleteCat"];
$catType = $_REQUEST["catType"];
$catDesc = $_REQUEST["catDesc"];
$editCat = $_REQUEST["editCat"];
$editCatInsert = $_REQUEST["editCatInsert"];
$newCatDate1 = $_REQUEST["newCatDate1"];
$newCatDate2 = $_REQUEST["newCatDate2"];

```

V jakém módu se tedy zobrazí například okno pro editaci obsahu, určí série podmínek, která následuje. Výsledky jsou ukládány do pole superglobální (nevyžaduje registraci) proměnné `$_SESSION`. Toto řešení totiž funguje i v prostředí, kde je zakázán příkaz `register_globals` (od verze PHP 5.0.0).

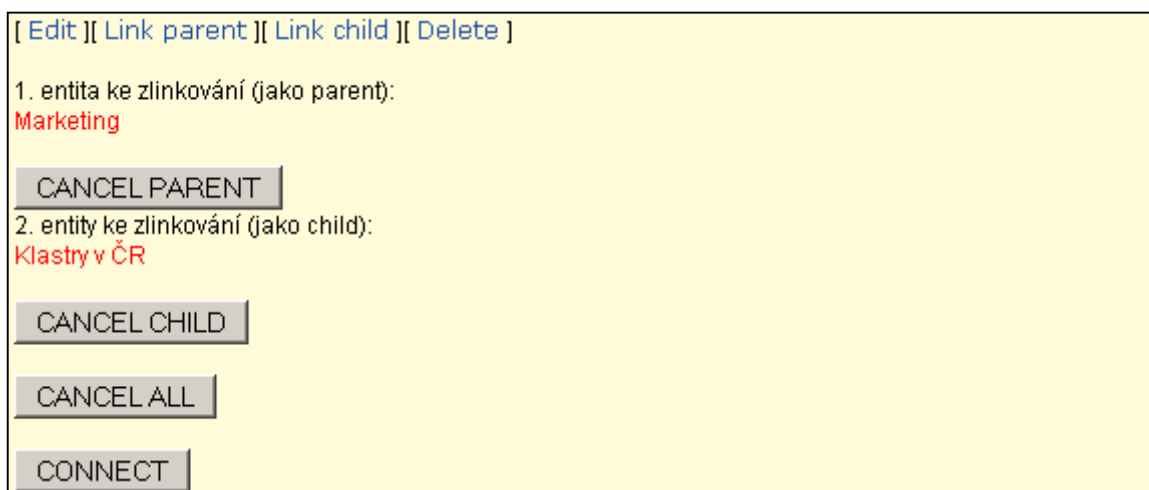
```

if ($_REQUEST["linkCatParent"])
{
    if ($_REQUEST["linkCatParent"] != "cancelParent")
    {
        $_SESSION[parentLink] = $_REQUEST["linkCatParent"];
        $_SESSION[nameParentLink] = zjist_jmeno_entity($_REQUEST["linkCatParent"]);
    }

    else if ($_REQUEST["linkCatParent"] == "cancelParent")
    {
        $_SESSION[nameParentLink] = "";
        $_SESSION[parentLink] = "";
    }
}

```

Výše uvedený kód zkoumá, zda proměnná `linkCatParent` obsahuje vůbec nějakou hodnotu. Když ano, zjistí jaké jméno Entity odpovídá předané hodnotě. Pokud se pak proměnná `linkCatParent` rovná `cancelParent`, proměnné zodpovědné za propojování entity v roli rodiče se vyprázdní a v tomto směru editace končí. Stejný postup je zvolen i pro propojování entit v roli potomka i pro hromadné ukončení celého propojování pomocí tlačítka CANCEL ALL.



Obr. 10 Mód propojování

Ať je však mód editace jakýkoliv, vždy je nutné zachovat možnost navigace. Ta je řešena na principu „stromu“, který se automaticky generuje podle obsahu databáze při každém zobrazení stránky. Algoritmus prochází postupně „pavouka“ navigace, zkoumá, zda jsou jednotlivé uzly rozbaleny, a podle toho je zobrazuje. Informace o tom, jestli je uzel rozbalený nebo ne, je uložena v superglobální proměnné, která má tvar pole. Toto pole vypadá takto –

`$_SESSION[stateKnot] [ID_uzlu] .`

```
if ( $_REQUEST[ "expand" ] ) $_SESSION[ stateKnot ][ $_REQUEST[ "expand" ] ] = "knotExpanded";
if ( $_REQUEST[ "contract" ] ) $_SESSION[ stateKnot ][ $_REQUEST[ "contract" ] ] = "knotContracted";
```

V uvedené části kódu se kontroluje, zda nedochází předáním superglobální proměnné `expand` nebo `contract` ke změně stavu některého z uzlů, a pakliže ano, je tento stav patřičně změněn. Z databáze se pak zjistí jméno uzlu, které ještě projde řadou úprav formátu. Je zkráceno na maximální délku 18ti znaků, vybaveno odpovídajícím hypertextovým odkazem, odsazeno mezerami podle toho, v jaké úrovni navigace se nachází, a teprve potom zobrazeno.

Jinak je způsob výpisu z databáze vždy podobný a vypadá principiálně takto:

```
 $sql = "select category.name, category.idcat, category.type FROM category,
        link WHERE (link.idChild ='$idKnot' and category.idcat=link.idParent) ";
 $result = mysql_query( $sql );
```

Nejdříve je proveden SELECT, který je následně uložen do proměnné \$result. Je-li výsledkem selectu jednorozměrné pole (tedy jen prvky v rámci jednoho řádku), jeho prvky lze adresovat jednoduše pomocí \$result ["jmeno_prvku"]. Jestliže se jedná o řádků víc, je nutné tyto řádky projít postupně, například za použití tohoto cyklu WHILE, který je opakován pro všechny řádky uložené v \$result:

```
while ( $rs = mysql_fetch_array( $result ) )
(
    $rs[ "name" ] = osekej_nazev( $rs[ "name" ] );
    $imageSrc = '';
    $actual_link = '<a href="index.php?idKnot='.$rs[ "idcat" ]. '&showIndex='
        . $_GET[ "showIndex" ]. '>'. $rs[ "name" ]. '</a>';

    echo ( '<tr valign="middle"><td width="16px" align="left">'. $imageSrc. '</td>
        <td width="260px" align="left" class="uzelData">'. $actual_link. '</td></tr>' );
)
```

Kompletní stručně okomentovaný kód je na příloženém CD.

13 Grafický layout

Základním cílem grafické stránky portálového řešení je zajistit, aby se s ním uživateli dobře pracovalo a podvědomě se nebránil jej navštěvovat. Jaké cesty vedou k tomuto cíli?

- 1- přehlednost** Musí být vždy okamžitě jasné, co je kde zobrazeno.
- 2- intuitivní ovládání a navigace** Nelze očekávat, že si budou uživatelé číst nějaké obsáhlé návody nebo příručky pro zacházení s portálem. Maximem je podle mého názoru 1-2 stránkový stručný výklad, ovšem v ideálním případě by měli i základně počítačově gramotní uživatelé sami od sebe rychle pochopit, jak s portálem zacházet a kde co najít. Důležité je umístění funkčních prvků tam, kde to uživatel čeká – podle jednoho z průzkumů⁶ je 37% prvků na 4/5 stránek řešeno stejným způsobem. Sem patří například logo v levém horním rohu a formulářové okénko pro vyhledávání na hlavní stránce – tyto konvence je užitečné dodržet.

⁶ Jakob Nielsen, The need for web design standards

- 3- grafický design** Velmi diskutabilní oblast. Každý má jiný vkus a zde rozhodně univerzálně vhodné řešení neexistuje. Důležité je proto dodržet alespoň obecně uznávané principy – používání méně sytých barev, které neunaví oči. Je také dobré vyhnout se prvkům obsahujícím animaci a hlídat barevné kombinace, které mohou být špatně čitelné. Podobných pravidel je celá řada, ovšem jde především o užitnou hodnotu a snadnost ovládní, nikoliv o originalitu a neotřelost. Zároveň by ale portál neměl být vyloženě fádni – rozhodně se vyplatí použití pěkně navržených grafických symbolů v navigaci a obecně v rozumné míře obrázků a dalších grafických prvků.
- 4- rychlost** V době narůstajícího podílu vysokorychlostního internetu má mnoho tvůrců stránek tendenci umísťovat na ně čím dál víc datově objemnější prvky. Tento přístup je sice částečně pochopitelný, avšak v řadě případů chybný. Podle údajů ČSÚ za rok 2005 ⁷ je například kolem 20% malých a velmi malých podniků (tedy těch, o které se v případě klastrů jedná především) připojeno pomalejším připojením než 144 kbit/s a 10% ze všech podniků pro připojení dokonce stále využívá analogový modem. Podle měření serveru zive.cz ⁸ pak činila průměrná rychlost mobilního připojení přes GPRS 45kbit/s a přes EDGE 85kbit/s (mobilní připojení, které je často využíváno manažery a zaměstnanci, navíc silně závisí na síle signálu, takže v některých případech může být dosahovaná rychlost skutečně velmi nízká). Co z toho vyplývá? Že řada uživatelů má stále velmi pomalé připojení, s čímž je nutno počítat a důkladně sledovat průměrnou velikost zobrazené stránky a také striktně omezit maximální velikost zobrazovaných grafických prvků. Má-li stránka velikost 100Kb, její stahování bude přes analogový modem či GPRS trvat minimálně 13 vteřin, což je pro řešení typu znalostního portálu nepřijatelné. Moje řešení dosahuje průměrné velikosti kolem 30Kb. To znamená kolem 4 sekund čekání u pomalého připojení, což už je výrazně lepší, s vypnutými obrázky se lze dostat na průměr kolem 18Kb, ovšem potom přestává být stránka použitelná. Pro zajímavost uvádím následující údaje:

⁷ ČSÚ, ICT 5-01 2006

⁸ Jan Šindelář, Speed test v květnu: výsledky rychlosti připojení

Adresa	Velikost titulní stránky	Doba downloadu při rychlosti 8Kb/s
seznam.cz	119 Kb	15 s.
atlas.cz	229 Kb	29 s.
Gogole.cz	4 Kb	0,5 s.
o2.cz	126 Kb	16 s.
jizdnirady.cz	146 Kb	18 s.
centrum.cz	308 Kb	39 s.

Tab 1. Průměrná doba downloadu, Měřeno 18.12.2006 ve 23:30

7- spolehlivost Uživatel musí vědět, že portál bude kdykoliv k dispozici, a to bez výpadků. Je nutné zajistit spolehlivý hosting včetně pravidelného monitoringu. To v mém případě nebylo ještě nutné řešit do detailů, zvolil jsem hosting u společnosti pipni.cz .

Jak splňuje výše uvedené požadavky moje řešení? Z části určitě ne, ale bylo by potřeba udělat alespoň omezený průzkum mezi potenciálními uživateli, abych si dovolil vyslovit nějaké objektivní závěry.

Co se technické stránky řešení týče, pracoval jsem na něm průběžně od 4. fáze projektu. Snažil jsem se proto design pojmout pokud možno minimalisticky, za hlavní cíl jsem si vytyčil přehlednost. V souboru styl2.css jsem vytvořil sadu stylů, které jsem přiřadil prvkům portálu. V tomto místě jsem se přitom dostal asi do největších obtíží během práce na portálu. Jakmile totiž dosáhla kombinace stylů v rámci tabulek a „podtabulek“ určité složitosti, začal se celý design hroutit a stávat nepoužitelným. Styly jsem nakonec vytvářel třikrát od začátku, přičemž jsem pokaždé rezignoval na část svých požadavků. Skoro stejný čas jsem potom strávil ve snaze zajistit, aby design vypadal stejně v obou nejpoužívanějších prohlížečích⁹, Internet Exploreru a Firefoxu. To se mi nakonec úplně nepodařilo, design se v obou prohlížečích mírně liší, mým představám se více blíží zobrazení v Mozille.

⁹ W3 Schools, Browser statistics

Práci nad designem jsem celkově strávil minimálně týden, ovšem výsledek moje očekávání nesplnil. Design není barevně sladěný a nejhorším nedostatkem je, že nakonec není ani příliš přehledný.

Práce na designu portálu byla tedy tou nejméně produktivní. Důvod je zřejmý – vytvořit atraktivní a přehledný design je mnohem intuitivnější činnost než např. programování funkcí. Postup se nedá najít v dokumentaci a vyžaduje jistou dávku výtvarného citu v kombinaci s perfektní orientací v problematice kaskádových stylů, DHTML a jednotlivých prohlížečů. Bez těchto znalostí se práce stává značně frustrující a neúčelná.

14 Shrnutí vývoje portálu

Petr Paleta ¹⁰ uvádí 6 základních příčin problémů softwarových projektů, pokusím se na závěr zhodnotit, jak jsem se s těmito riziky vypořádal, a to i přesto, že se v mém případě o úplně klasický softwarový projekt nejednalo.

- 1. Zpoždění** Na práci jsem měl pevně vymezený čas (cca. 6 týdnů) a mým úkolem bylo odevzdat to, co stihnu, takže zpoždění v tomto případě v podstatě nehrozilo
- 2. Vysoká chybovost** Kód jsem odlaďoval v celém průběhu projektu, nicméně jen v nejnútnejší míře, jednalo se přece jenom o prototyp a chtěl jsem spíše předvést co největší funkčnost, byť s možnými chybami, než nějaké perfektně odlaďené torzo zamýšleného projektu, které ale nebude tím pádem zřejmě nikdy dále využito.
- 3. Nesplnění požadované funkčnosti** Požadovaná funkčnost byla vskutku jen základní (zobrazení a editace), nesplnit ji by znamenalo nic nenaprogramovat, takže zde jsem obavy neměl.
- 4. Nedostatečná výkonnost** Jak už jsem se zmínil dříve, datový rozsah projektu byl tak malý, že na výkon enginu jsem nemusel nijak dbát.
- 5. Složitě uživatelské rozhraní** V tomto bodě byla situace složitější a z důvodů popsaných v kapitole 13 zde moje řešení zcela neobstálo, avšak pro demonstrační účely dostačovalo.
- 6. Obtížná udržitelnost programu** Problematiku udržitelnosti jsem neřešil vůbec. Komentář jsem nevytvářel prakticky žádný. Vycházel jsem z toho, že pokud se moje řešení líbit nebude, zapomene se na něj a už ho nikdy nebude potřeba nijak rozvíjet ani udržovat. V případě, že by naopak mělo řešení úspěch a bylo by rozhodnuto o jeho dalším využití, musel by se vývoji věnovat

jistě menší tým a nejjednodušší by pak bylo zřejmě všechno naprogramovat znovu a mnohem důkladněji.

Jak je z předchozího shrnutí patrné, mnoho rizik odpadlo už kvůli samotné povaze projektu a těm zbylým jsem se více či méně úspěšně vyhnul. Základním měřítkem úspěchu projektu bude nakonec rozhodnutí, zda se bude na položených základech pokračovat, nebo jestli vývoj začne znovu, tentokrát jiným směrem (toto rozhodnutí mi bohužel v době dokončování bakalářské práce ještě nebylo známo).

Na závěr ještě jeden postřeh – čím více jsem byl do práce na projektu a do různých, místy téměř akademických, debat okolo něho ponořen, tím méně jsem byl schopen dívat se na celou věc očima uživatele, pro kterého teoreticky projekt vznikal. Tento uživatel totiž nejen že nebude znát žádné teorie kolem map námětů, ale nejspíš ani nebude vědět, co to mapy námětů jsou. Stejně tak tomu bude u zbytku teoretického aparátu, na kterém je portál budován. Ta největší výzva potom je tyto teoretické základy přetavit ve fungující systém tak, že i uživatel bez jejich nejmenší znalosti je bude nevědomky využívat, a tak z nich těžit ve svůj prospěch.

15 Znalostní portál a jeho komerční nasazení

Můj projekt byl pouze skromným prototypem, při jeho tvorbě jsem se ale stále zabýval myšlenkou, co všechno by měl „opravdový“ znalostní klastrový portál obsahovat a jak by měl vypadat. Některé z těchto námětů jsou popsány v kapitole 2.3 a podobných vylepšení se dá najít nepřeborná řada, ovšem práce na takovémto projektu by již vyžadovala tým minimálně 2-3 lidí (programátor, webdesigner/grafik/kodér), jeho vývoj by trval řádově měsíce a náklady by se tak po čase mohly dotknout i milionové hranice. Je potom otázka, zda by si jeden samotný klastr byl ochoten takový software zaplatit. Klastrových iniciativ ale v ČR vznikají v poslední době desítky, a tak by možným řešením bylo vyvinout a odladit špičkový engine, který by se poté vždy pouze upravil podle potřeb konkrétního

¹⁰ Petr Paleta, Co programátory ve škole neučí

klastru. Konečná cena řešení pro zákazníka by se tak mohla pohybovat řádově ve statisících a to už je částka, kterou by manažeři některých klastrů zřejmě byli ochotni zaplatit – vždyť součástí grantu EU je mimo jiné suma cca. 3 000 000 Kč na pořízení potřebných hardwarových a softwarových technologií.

Pokud by se systém komerčně ujal, dal by se samozřejmě dále rozvíjet. Uvažovat lze o prostředí pro přímou hlasovou a obrazovou komunikaci, sdílení velkých objemů dat pro výzkumné a obchodní účely, moduly pro on-line výzkumnou spolupráci. Možností je zde velké množství a mnoho z nich ještě nikdy nebylo vyzkoušeno v praxi, jedině kde se často ukáže, že i zdánlivě neperspektivní myšlenka může mít mezi uživateli velký úspěch.

16 Závěr

Poslední fáze mého studia na Vyšší odborné škole informačních služeb mi přinesla jedinečnou a nenahraditelnou zkušenost. V rámci praxe jsem měl možnost se v prostředí špičkové softwarové firmy samostatně zabývat zadaným projektem. A protože se jednalo o součást grantu, i v prostředí zkušených odborníků s dlouholetou praxí jsem měl možnost přispět svým dílem při hledání nových originálních řešení, což byla zřejmě nejvíce motivující část mé zkušenosti jak ve vztahu k projektu samotnému, tak ve vztahu k projektům budoucím. Co jsou ty nejdůležitější poznatky? Základem každého projektu je co nejširší debata se zadavateli. Je nutné co nejlépe pochopit, jaké mají očekávání, a toto očekávání se pokusit naplnit. Jsou-li potom jejich představy mlhavé, je dobré pokusit se vydat směrem, o kterém lze předpokládat, že je správný, a co nejčastějšími konzultacemi tento směr upřesňovat. Pokud tato spolupráce zafunguje, nejsou to už jen zadavatelé, kdo vám ukazuje cestu, ale zároveň vaše vlastní tvůrčí práce ukazuje cestu jim. Když potom debata o dalším postupu nikam nevede, je někdy lepší spolehnout se na intuici a pokusit se o řešení „na vlastní pěst“. To sice občas znamená zbytečnou práci, ale v mnoha případech naopak cestu ze slepé uličky. Zásadní věcí se také často ukázalo umět výsledky svojí práce náležitě popsat a vysvětlit. Na to však bohužel často nebývá dostatek času, a potom je tedy alespoň nutné snažit se zaujmout atraktivním zpracováním a designem. Čím více tato formální stránka přitáhne pozornost těch, kteří rozhodují o budoucnosti projektu, tím větší je naděje, že se budou zabývat i stránkou obsahovou a třeba i náležitě ocenit a využít odvedenou práci.

Další poznatek je, že naopak nemá cenu být přehnaně pečlivý a zdržovat se maličkostmi. Často se jedná o obtížně dohledatelné výjimky a záludnosti jednotlivých systémů a aplikací, které nemají zásadní význam, a jejich řešení zabere hodiny i dny. Řešit je se vyplatí až při případném praktickém nasazení produktu a lze na to najmout povolané odborníky, kteří když odvedou kvalitní a rychlou práci, prostředky i přes obvykle vysoké finanční požadavky nakonec spíše uspoří.

Co říci závěrem? Na popisovaném svěřeném projektu jsem pracoval rád a i z mojí práce je snad poznat, že jsem mu věnoval maximální možnou pozornost, byť jsem se musel zároveň prakticky všechny potřebné postupy doučit, či si je přímo nastudovat od začátku, což je ale při praxi věc pochopitelná. Výsledek také není jistě ideální, ale odevzdal jsem funkční prototyp systému, na jehož vývoji jsem měl tvůrčí podíl, který snad moje práce dostatečně výstižně popisuje.

17 Seznam použitých zdrojů

1) Czechinvest, Klast. Dostupný z www:

<<http://www.czechinvest.cz/web/pwci.nsf/pages/AAFE16F7D4C99736C1256EBC00589B97?OpenDocument>>

1) RAMANUJAM, T. C. A. . Into the fourth dimension of economics . Business line [online]. 2006 [cit. 2006-12-20]. Dostupný z WWW:

<<http://www.thehindubusinessline.com/2006/08/18/stories/2006081800891000.htm>>.

2) IEC 13250:2003 Dostupný z WWW :

<http://www1.y12.doe.gov/capabilities/sgml/sc34/document/0129.pdf>

3) NSS group white paper, 2006 [cit. 2006-12-18]. Dostupný z WWW:

<http://www.nss.co.uk/WhitePapers/CorporatePortals.htm>

4) JP, PHP login script with remember me feature. 2004. Dostupný z WWW:

<http://www.evolt.org/article/comment/17/60265/index.html>

5) NIELSEN, Jakob. The Need for Web Design Standards. Jakob Nielsen\'s Alertbox [online]. 2004 [cit. 2006-12-21]. Dostupný z WWW:

<http://www.useit.com/alertbox/20040913.html>

6) ČSÚ, ICT 5-01 2006 [online] 2006 [cit. 2006-12-16] Dostupný z WWW:

http://www.czso.cz/csu/redakce.nsf/i/podnikatelsky_sektor

7) SpeedTest v květnu: výsledky rychlosti připojení. Živě.cz [online]. 2006 [cit. 2006-12-18]. Dostupný z WWW:

<<http://www.zive.cz/h/Uzivatel/AR.asp?ARI=129763&CAI=2114>>.

8) w3 Schools, Browser statistics. [online]. 2006. . Dostupný z WWW:
<http://www.w3schools.com/browsers/browsers_stats.asp>

9) PALETA, Petr. *Co programátory ve škole neučí*, 2003. 337 s. ISBN 80-251-0073.

18 Přílohy

CD s kompletním zdrojovým kódem portálu. Portál ve funkční podobě je přístupný na <http://klastr.zewl.com>

Uživatelské jméno: admin

Heslo: nimda