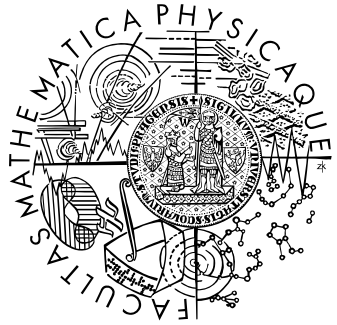
Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Kristýna Knapová

Seskupování zpráv

Středisko informatické sítě a laboratoří

Vedoucí bakalářské práce: Mgr. Jakub Vrána

Studijní program: Informatika, Obecná informatika

2009

Děkuji Mgr. Jakubovi Vránovi za vedení mé práce a odborný dohled.

Prohlašuji, že jsem svou bakalářskou práci napsala samostatně a výhradně s použitím citovaných pramenů.

Souhlasím se zapůjčování, práce a jejím zveřejňováním.

Kristýna Knapová

V Praze dne: 31. 7. 2009

Obsah

Název práce: Seskupování zpráv

Autor: Kristýna Knapová

Katedra (ústav): Středisko informatické sítě a laboratoří

Vedoucí bakalářské práce: Mgr. Jakub Vrána

e-mail vedoucího: jakub@vrana.cz

Abstrakt: Cílem předložené práce je vytvoření webové aplikace, která bude poskytovat nejnovější zprávy v přehledné podobě. Zprávy budou umístěné do skupin podle stejného tématu a seřazené podle zajímavosti, důležitosti a aktuálnosti.

Klíčová slova: Zprávy, Seskupení, Shluková analýza, Google

Title: News Feed Clustering

Author: Kristýna Knapová

Department Network and Labs Management Center

Supervisor: Mgr. Jakub Vrána

Supervisor’s e-mail address: jakub@vrana.cz

Abstract: The aim of this work is a creating of the web application that will provide user with up-to-date news in a well-arranged list. The news will be categorized within groups with the same subject and ordered by the matter of interest, importance and topicality.

Keywords: News, Grouping, Clustering, Cluster analysis, Google

1 Úvod

V dnešní době nás svět informací a internetu pohlcuje. Lze ovšem takové množství dat získat rychle a bez zdlouhavého čekání? Může je někdo nebo něco vyhledávat za nás? Jak poznat, co je vlastně důležité a co nikoliv? A co když se ty informace opakují, jak je publikovat pouze jednou? Lze naprogramovat takového robota, který toto všechno provede za nás?

Nabízí se tedy hlavní otázka, je možné veškerý objem nových informací pořizovat automaticky bez lidského zpracování a publikovat je ve srozumitelné podobě? Ve svém projektu bych se zaměřila na vyhledávání a především seskupování zpráv podle stejného tématu. Existuje spousta zpravodajských serverů (tisíce, desetitisíce a každým dnem jich pravděpodobně přibývá) a kdybychom je měli všechny stále prohledávat, tak to jediný člověk nikdy nemůže stihnout. Navíc zprávy se opakují, každou minutou přibývají nové nebo se rozšiřují o nová fakta. Později by se ukázalo, že je to Sysifovská práce.

Bakalářská práce bude speciálně zaměřená na seskupování zpráv, které řeší stejné téma nebo reagují na tutéž událost. Užitečnost tvorby podobných skupin je známá i z reálného života. Podíváme-li se na objem informací, který se v dnešní době na internetu vyskytuje, jde o takové množství dat, které už nelze v celém svém obsahu prozkoumat. Většina uživatelů začíná na vyhledávacích portálech, aby nalezli potřebné údaje, nebo využívá systematicky uspořádaných katalogů seskupených podle stejného zaměření stránek.

Vyhledávání bude probíhat na největších zpravodajských serverech, kde přibývají zprávy téměř každou minutu a je třeba pořízené informace okamžitě zpracovat a publikovat. Bohužel ale nejsou k dispozici výkonné počítače, které by mohly proces urychlit. Implementace musí běžet na klasickém stolním PC.

Při tvorbě práce sloužila jako ukázková aplikace Google news, která je ekvivalentem výsledného řešení, ale využívá úplně jiné možnosti. Hlavně rozložení výkonu aplikace je ohromná výhoda. Distribuované systémy, které má Google k dispozici, nabízejí jiné možnosti implementace, a proto je podobný pouze výsledek, nikoliv řešení.

Bakalářská práce nejdříve obsahuje teoretický úvod do problematiky. Následuje nejdůležitější část, která se věnuje rozboru samotného řešení a vysvětlení implementace. Poslední část je věnovaná analýze, návrhu celé aplikace a popis databáze.

2 Úvod do problematiky

S rozpoznáváním podobnosti mezi objekty nebo s tříděním objektů do skupin se můžeme setkat v běžném životě celkem často. Pokud je třeba tuto činnost automatizovat nebo algoritmizovat, je to již složitější problém. V odborné literatuře se o problematice mluví jako o shlukování nebo shlukové analýze (clustering, cluster analysis).

2.1 Shluková analýza

Jedná se o oblast řešení, kdy se objekty systematicky třídí do skupin. Podíváme-li se na příklad z vlastní zkušenosti, můžeme celý problém aplikovat na svoje věci, které třídíme, abychom se v nich vyznali. Například oblečení máme ve skříních uloženo podle nějaké základní společné charakteristiky (všechna trička, ponožky, kalhoty jsou v samostatných regálech). Čím více oblečení budeme mít, tím je potřeba třídění užitečnější a nutnější.

Stejné myšlenky využíváme při organizaci dat na disku, abychom co nejdříve našli potřebnou informaci. Jedná se tedy o starou myšlenku a následující autoři se snažili o vytvoření základní charakteristiky nebo definice.

R. C. Tryon (1939): *„Shluková analýza je obecně logický postup formulovaný jako procedura, pomocí níž seskupujeme objektivně jedince do skupin na základě jejich podobností a rozdílností.“*

R. E. Bonner (1964): *„Je dána množina objektů, z nichž je každý definován pomocí množiny znaků s ním souvisejících. Tato množina znaků je pro každý objekt stejná. Máme nalézt shluky objektů (podmnožiny původní množiny objektů) tak, aby si členové shluku byli vzájemně podobní, ale nebyli si příliš podobní s objekty mimo tento shluk.“*

M. R. Anderberg (1975): *„Tento problém je obvykle charakterizován jako hledání přirozených skupin. Konkrétněji jde o třídění pozorování do skupin tak, aby stupeň přirozené asociace členů téže skupiny byl vyšší a členů různých skupin nižší.“*

2.1.1 Metody shlukové analýzy

Metody shlukové analýzy lze rozdělit podle 3 faktorů.

Metody dle cíle shlukování

- nehierarchické, produkující prostý rozklad objektů na podmnožiny
- hierarchické, produkující hierarchii rozkladů, kde každý rozklad je zjemněním předcházejícího

Metody dle typu výsledných shluků

- shluky kulové, body soustředěné kolem svého těžiště
- shluky obecné tvoří souvislé husté oblasti nejrůznějších tvarů

Metody dle typu rozkladu

- shluky disjunktní
- shluky překrývající se

2.1.2 Faktory, které ovlivňují skupiny

Většinou je být dost obtížné přesně definovat, jak má skupina vypadat. Pro nejlepší výsledek je třeba vyřešit následující problémy:

- Výběr atributů reprezentující podobnost.
- Koeficient podobnosti skupin (shluků).
- Počáteční rozklad, obecná skupina.
- Pojem vzdálenosti skupin.

Výběr atributů reprezentující podobnost

Základem pro každou skupinu, je definice podobnosti. Nejdůležitější je uvědomit si, jaké atributy signalizuji podobnost, jestli jich je více nebo jeden, jak jsou důležité.

Koeficient podobnosti skupin (shluků)

Tento koeficient je většinou dost obtížné určit. Je potřeba nalézt hranici, kde všechny prvky, které ji splňují, spadají do skupiny, a všechny prvky, co jsou mimo hranici, již do skupiny nepatří. Pokud se nalezne hranice, kdy toho rozdělení funguje na 100, je to nejlepší možné řešení. V praxi však taková úspěšnost není. Záleží ovšem na jednotlivých případech.

Počáteční rozklad, obecná skupina

Počáteční rozklad nebo definice obecné skupiny, může zrychlit proces rozpoznávání. Jedná se o jakousi přípravu, která potom pomůže v dalším zpracování. Většinou lze alespoň definovat rozklad pro vytvoření potřebných atributů, které slouží k rozdělení do skupin.

Pojem vzdálenosti skupin

Tímto pojmem lze vyřešit, jak moc mají být skupiny odlišné, jestli mohou být mezi sebou podobné nebo se mohou překrývat.

2.2 Algoritmy shlukové analýzy

2.2.1 Pro nehierarchické shluky, K středová metoda

K-středové metody optimalizační hledají nejlepší rozklad množiny objektů iteračním způsobem.

Počáteční rozklad (zadaný nebo vygenerovaný) zlepšují tak, že hledají rozklad s lepší hodnotou kriteriální funkce.

Algoritmus

1. zadání počátečních typických k bodů
2. přiřazení každého bodu k nejbližšímu typickému bodu a jemu odpovídajícímu shluku
3. výpočet těžiště každého z k-shluků
4. definování nových typických bodů ve vypočtených těžištích
5. pokud došlo ke změně v přiřazení bodů shlukům, opakování od bodu 2
6. výpočet kriteriální funkce výsledného rozkladu

2.2.2 Pro hierarchické shluky

Algoritmus

1. výpočet matice podobnosti objektů, počáteční rozklad tvoří jednoobjektové shluky
2. nalezení nejmenší vzdálenosti shluků v aktuální „hladině“ hierarchie
3. spojení těchto nejbližších shluků do společného shluku vyššího stupně hierarchie, ostatní shluky zůstanou nezměněny
4. výpočet charakteristik shluků aktuální hladiny rozkladu
5. pokud existuje více než 1 shluk, opakování od bodu 2.

3 Dobývání informací

RSS zprávy je třeba rozdělit do skupin podle stejného tématu. Nejdůležitějším faktorem je shoda v klíčových slovech jednotlivých článků a nalezení vhodné míry podobnosti.

3.1 Získání klíčových slov

Pro získání klíčových slov je možné využít dva přístupy.

3.1.1 Existující slovník

První z nich vezme již existující slovník a všechna slova si uloží do své tabulky. Následně by se využívala slova, která by se pořídila pouze tímto způsobem.

Výhody

- Již na začátku máme všechna potenciální klíčová slova, která se mohou objevit v článcích.
- Dopředu je přibližně jasná mohutnost celé tabulky.
- Jelikož do tabulky nebudou přibývat další údaje, dalo by se pracovat s významem slov a skloňováním či časováním a naprogramovat nad ní lemmatizátor.

Nevýhody

- Nalezení takto kompletního slovníku.
- Nově vznikající slova, převzatá slova, nové termíny nebo technologie nebudou součástí slovníků.
- Identifikace často používaných slov je problematická.

3.1.2 Vlastní slovník

Vlastní slovník bude postupně vznikat z článků, které se získají z RSS zdroje.

Výhody

- Není třeba hledat již vytvořený slovník.
- Slovník se bude stále rozšiřovat a aktualizovat.
- Při přidávání slov je možné sledovat četnost výskytu slova a lze oddělit často používaná slova, jako jsou předložky a spojky.
- Je možné vytvářet více slovníků různých jazyků.

Nevýhody

- Pomalejší vytvoření kompletního slovníku.
- Implementace lemmatizátoru závislá na době vytváření slovníku.

3.1.3 Implementace klíčových slov

Výsledné řešení využívá tvorbu vlastního slovníku a vzniká s pořizováním obsahu každého článku.

Každý článek nejdříve zpracuje aplikační funkce ve třídě ItemRSS, která dostane titulek a obsah zprávy a rozdělí všechna slova z článku do pole.

Porušení typografických pravidel nebo sjednocení formátu

Je nutné brát v úvahu, že spoustu článků nedodržuje základní typografická pravidla a mohou tak způsobit odlišnost stejných slov. Jedná je hlavně o problém tečky, čárky a dvojtečky, které by měli mít za sebou mezeru.

Další problém může nastat při formátování textu, například pří zobrazení čísel. Tyto problémy by měly být odstraněny už v průběhu zpracování slov. Následující text rozebírá podrobněji způsob opravy nebo sjednocení formátu.

Porušení tečky

Je třeba odchytnou případy souvislého textu bez mezery za tečkou, ale naopak neoddělovat webové odkazy a čísla s desetinnou čárkou.

Potřeba rozdělení

Jedna věta.Další věta. => Jedna věta. Další věta.

Není třeba rozdělení

Číslo 15.5 => Číslo 15.5

Porušení čárky

Je třeba odchytnou případy souvislého textu bez mezery za čárkou, a opravit případy, kdy je čárka použita pro oddělení desetinných míst.

Potřeba rozdělení

Jedna věta,další věta. => Jedna věta, další věta.

Potřeba opravy

Číslo 10,6 => Číslo 10.6

Porušení dvojtečky

Je třeba odchytnou případy souvislého textu bez mezery za tečkou. Naopak hodnoty času musí zůstat pohromadě.

Potřeba rozdělení

Jedna věta:další věta. => Jedna věta: další věta.

Není třeba rozdělení

Čas 15:50 => Čas 15:50

Porušení vykřičníku

Je třeba odchytnou případy souvislého textu bez mezery za vykřičníkem.

Potřeba rozdělení

Jedna věta!Další věta. => Jedna věta! Další věta.

Porušení otazníku

Je třeba odchytnou případy souvislého textu bez mezery za otazníkem.

Potřeba rozdělení

Jedna věta?Další věta. => Jedna věta? Další věta.

Sjednocení formátu čísel

Sjednocení formátu čísel je další nutnou potřebou. Pro formátování dlouhých čísel se oddělují tisíce mezerou kvůli čitelnosti.

Je třeba sjednocení

Výplata činila 20 000. => Výplata činila 20000.

3.1.4 Funkce getWordArray

Funkce dostane text a rozdělí všechna slova do pole, kdy se snaží vyřešit všechny předchozí problémy v typografických pravidlech.

Funkce implementuje následující algoritmus:

1. Vrať všechna slova oddělená novou řádkou, tabulátorem nebo mezerou do pole.
2. Vytvoř pole všech znaků (`":"`, `"?"`, `"!"`, `"."`), které mohou porušit typografické pravidlo rozdělení
3. Pro každé slovo ze získaného pole proveď následující postup:
    a. Odstraň následující speciální znaky - `:.,?!'[]()´"„“’‘;-…` ze začátku a konce slova.
    b. Vezmi slovo, pouze pokud není odkazem (neobsahuje podřetězec http) a současně má délku dva a více znaků nebo je to pouze jeden znak, buď číslo anebo předložka či spojka.
    c. Ošetři pravidla pro čárku.
    d. Ošetři pravidla pro čísla oddělená mezerou.
    e. Ošetři pravidla pro ostatní speciální znaky, které jsou připravené v poli.
    f. Ulož slovo do pole.
4. Vrať všechna získaná slova.

Ošetření pravidla pro čárku je provedeno následovně:

1. Dokud pozice čárky ve slově je někde uprostřed slova, prováděj následující postup.
2. Pokud není znak před čárkou číslo a současně znak za čárkou číslo, rozděl slovo na dvě samostatná.
3. Pokud jsou znaky před čárkou a za čárkou čísla, nahraď čárkou tečkou a ulož do pole.

Ošetření pravidla pro číslo je provedeno následovně:

1. Pokud slovo obsahuje pouze čísla, pokračuj dále kontrolou následujícího slova.
2. Dokud budou následující slova obsahovat pouze čísla, připojuj jej k předchozímu slovu.
3. Současně zvyšuj index právě procházeného slova.
4. Jakmile již nebude následovat pouze číslo, ulož nové slovo do pole.

Ošetření pravidla ostatních znaků je provedeno následovně:

1. Pro každý speciální znak, zkus najít pozici ve slově.
2. Pokud pozice bude uprostřed slova, pokračuj dál.
3. Pokud není znak před čárkou číslo a současně znak za čárkou číslo, rozděl slovo na dvě samostatná.
4. Ulož nové slovo do pole.

Všechny nové zprávy využívají funkci **getWordArray**,aby získaly seznam slov, který uloží do speciálního atributu v databázové tabulce.

3.2 Získání obsahu RSS článku

Získání obsahu článku je spojeno s tvorbou vlastního slovníku a je již řešené na úrovní databázového systému pomocí uložených procedur.

Obecně je tento proces přípravy atributů klíčových slov velice důležitý. Je to třetí teoretický bod, který je třeba pro optimalizaci zpracování.

Články, které nově přibudou do databáze, mají výchozí hodnotu pro atribut kontent nastavenou na nulu. Tím je jasné, že článek musí projít rozebráním obsahu na klíčová slova a připravením slov pro rozpoznávání skupiny.

Aby nedocházelo k velkému zpracování dat najednou, začne se tvořit obsah pro posledních několik článku.

Následující dotaz vybere všechny zprávy, ze kterých se bude získávat obsah.

```
SELECT idItemRSS FROM itemrss i
WHERE content=0
ORDER BY pubdate LIMIT 1000;
```

Pro každé identifikační číslo se volá funkce **createKeywodsTableFromArticleById**, která vytváří slovník a obsah zprávy.

3.2.1 Tabulka keywordsall

Tabulka keywordsall v sobě uchovává slovník. Kromě názvu slova obsahuje počet dosavadních výskytů. Je to výhodné pro získání tzv. zakázaných slov, která by neměla být využívána jako klíčová. Jedná se především o spojky a předložky, které neřeší obsahovou stránku, ale pouze jako výrazové prostředky.

Sloupce tabulky

keywordsall 1

Key	Column Name	Datatype	Not Null	Default	Comment
PK	**idKeyword**	INT(10)	Yes		Číslo slova
	keyword	CHAR(40)	Yes		Slovo
	countkey	INT(10)	No	'1'	Počet dosavadních výskytů
	used	TINYINT(1)	Yes	'1'	Zakázané vs. povolené slovo

Indexy

Tabulka využívá dva unikátní klíče:

IdKeyword je primární klíč a identifikátor, který bude použít v tabulkách jako cizí klíč. V porovnávání obsahu se tak nebudou porovnávat názvy, ale pouze čísla.

Keyword zachovává vlastnost jedinečného identifikátoru, aby v tabulce nemohla být dvě shodná slova s jiným číslem. Unikátní index si tuto vlastnost bude hlídat. Navíc lze potom využít konstrukce ON DUPLICATE KEY UPDATE, pří vkládání nových hodnot.

keywordsall 2

Index Name	Columns	Primary	Unique	Type	Kind
PRIMARY	idKeyword	Yes	No	PRIMARY	
word	keyword	No	Yes	UNIQUE	BTREE

Vztahy (cizí klíče)

Tabulka keywordsall je ve vztahu s dalšími tabulkami, které využívají klíčová slova. Jedná se o tabulky s obsahem článků, obsahem skupin a zakázaná slova.

keywordsall 3

Relationship Name	Parent Table	Child Table	Card.
FK_articlecontentnew_keywordsall	keywordsall	articlecontentnew	1:n
FK_grouparticlecontentnew_keywordsall	keywordsall	grouparticlecontentnew	1:n
FK_keyworddisabled_keywordsall	keywordsall	keyworddisabled	1:n

3.2.2 Tabulka articlecontentnew

Tabulka articlecontentnew v sobě uchovává obsah článků, které ještě nemají rozpoznanou skupinu.

Sloupce tabulky

articlecontentnew 1

Key	Column Name	Datatype	Not Null	Default	Comment
PFK	idKeyword	INT(10)	Yes		Číslo slova
	countkey	SMALLINT(5)	Yes	'1'	Počet výskytů v článku
PK	idItemRSS	INT(10)	Yes		Číslo článku

Indexy

Primární klíč je spojený s kombinace atributů pro číslo slova a číslo článku. Každý článek může obsahovat více klíčových slov a stejně tak klíčové slovo ve více článcích.

Dále tabulka definuje index, který může obsahovat duplicity. Indexovaný je atribut idItemRSS, neboť se v tabulce bude vyhledávat jen pouze článku.

articlecontentnew 2

Index Name	Columns	Primary	Unique	Type
PRIMARY	idKeyword, idItemRSS	Yes	No	PRIMARY
idItemRSS	idItemRSS	No	No	INDEX

Vztahy (cizí klíče)

Tabulka obsahuje cizí klíč na klíčové slovo. Nastavuje referenční integritu a nastavuje způsob udržení referenční integrity. Pří aktualizaci i odstranění využívá CASCADE.

articlecontentnew 3

Relationship Name	Parent Table	Child Table	Card.
FK_articlecontentnew_keywordsall	keywordsall	articlecontentnew	1:n

3.2.3 Procedura createKeywodsTableFromArticleById

Procedura, která připraví a vytvoří obsah jednoho článku.

Algoritmus procedury:

1. Vezmi z článku atribut words, kde jsou připravená slova článku oddělená čárkou.
2. Postupně rozebírej jednotlivá slova, odstraň čárky a mezery.
3. Kontroluj, jestli není slovo zakázané (spojky, předložky, zájmena apod.).
4. V průběhu zpracování slov si průběžně vytvářej dotaz pro hromadný insert.
5. Pokud se nalezlo alespoň jedno slovo, pokračuj dál ve zpracování.
6. Nová slova se nejdříve vlož do dočasné tabulky, pokud jsou již v tabulce umístěny, využij konstrukce ON DUPLICATE KEY UPDATE a zvyš se počet výskytů.
7. Poté se slova ulož do tabulky keywordsall, kde jsou všechna klíčová slova. Pokud slova ještě neexistovala, dostanou své identifikační číslo. Takto bude jisté, že při vkládání do tabulky s obsahem, bude mít každý záznam svůj kompletní klíč složení s idKeywords a idItemRSS.
8. V dočasné tabulce s obsahem aktualizuj identifikační čísla klíčových slov z tabulky keywordsall.
9. Pokud se nalezlo více jak 10 slov, ulož data z dočasné tabulky do tabulky s obsahem a aktualizuj RSS zprávu, kde se nastav atribut kontent na 1.
10. Pokud se nalezlo méně jak 10 slov, nebude se rozpoznávat skupina a rovnou přiřaď univerzální skupinu 0, kam spadají všechny zprávy, kterým nelze skupinu určit.
11. Nakonec se odstraň dočasná tabulka.

3.3 Získání skupiny RSS článků

Získání skupiny článků je založeno na teorii shlukové analýzy.

3.3.1 Klasifikace shluku pro seskupení podle stejného tématu

Podle cíle shlukování půjde o nehierarchický shluk, produkující prostý rozklad objektů na podmnožiny. Nebudou džinové úrovně skupin.

Podle typu rozkladu jde o shluky disjunktní, každá zpráva pouze jedna skupina. Je žádoucí, aby jedno téma bylo pouze v jedné skupině.

3.3.2 Výběr atributů reprezentující podobnost

Atributem, který reprezentují podobnost, je klíčové slovo. Čím více klíčových slov mají dva články společné, tím je větší pravděpodobnost, že padají do stejné skupiny. Dalším důležitým faktorem je doba vydání článků. Tento atribut je důležitý zejména pro články podobného tématu, ale zaznamenávající již jinou událost.

Například sportovní zprávy o tenisovém turnaji Davis Cup mohly informovat o začátku turnaji, průběžných výsledcích i celkovém vítězi. Pokaždé šlo o jinou informaci, ale obsahovali podobná klíčová slova. Tím pádem by mohlo dojít k moc velkému shluku.

3.3.3 Koeficient podobnosti skupin (shluků)

Tento koeficient je většinou dost obtížné zjistit. Je vhodné průběžné výsledky monitorovat a potom z nich statistickými výpočty získat nejlepší možné nastavení.

Někdy je možné dospět k výsledku pouze sérií úvah a pouze otestovat na reálných datech. Je většinou vhodné vědět, kde začít.

Koeficienty, které při seskupování hrají roli, jsou:

- míra podobnosti článků a skupiny
- časový rozsah skupiny

Míra podobnosti (relevance)

Míru podobnost lze určit výpočtem:

```
relevanceActual = TRUNCATE(sumFoundKeywords / sumKeywords ,4);
```

Kde:

sumFoundKeywords *– je součet nalezených slov ze zprávy z nejvhodnější skupiny*

sumKeywords *– je součet všech slov zprávy*

TRUNCATE *– s parametrem 4 odtrhne desítitisíciny*

Výsledkem je tedy poměr mezi počtem nalezených slov a celkovým počtem.

Pozorováním se dospělo k nastavení, kdy doporučená relevance minimálně 0.2. Nižší hodnoty jsou na hranici, kdy se začíná tvořit shluk, který je příliš velký a obsahuje nesprávné rozdělení.

Čím vyšší relevance, tím vyšší pravděpodobnost, že je skupina správně vytvořena

Rozbor relevance.

Z celkového počtu 53466 zpráv lze vidět v tabulce Relevance 1, kolik zpráv mělo relevanci od 20 do 30 procent, od 30 do 40 procent, atd. 100% relevanci mají články, které tvoří skupinu.

Relevance 1

Relevance	Počet
Od 20 do 30 %	4241
Od 30 do 40 %	2580
Od 40 do 50 %	1436
Od 50 do 60 %	2040
Od 60 do 70 %	1719
Od 70 do 80 %	1512
Od 80 do 90 %	1449
Od 90 do 100 %	2156
100 %	36333

Problémy, které mohou nastat při určování koeficientu podobnosti.

Doporučená relevance může být příliš malá.

V souvislosti s tímto nastavením se může stát, že se do skupiny dostanou zprávy, které do ní nepatří.

Doporučená relevance může být příliš velká.

V souvislosti s tímto nastavením se může stát, že se do skupiny nedostanou zprávy, které do ní patří.

Časový rozsah skupiny

Dalším důležitým činitelem je časový rozsah skupiny. Tímto faktorem se dá provést optimalizace počtu potenciálních skupin, ve kterých se budou vyhledávat klíčová slova. Pomocí časového rozsahu zprávy lze vyřešit události o podobném tématu, které na sebe navazují.

Když se nad rozsahem zamyslíme, lze trochu vydedukovat, jak tento rozsah vytvořit. Pokud o nějakém aktuálním tématu, začne psát, jedná se jistě o čerstvou zprávu. Najde se tedy server, který danou informaci zveřejní první. Podle této zprávy se vytvoří nová skupina. Každý stejná zpráva s jiného zdroje již připadne do této skupiny.

Je třeba najít nejvhodnější interval od prvního vydání k poslednímu vydání. Předpokládejme, že většina serverů, které řeší nejnovější informace, bude o stejném tématu psát přibližně ve stejnou dobu. Z hlediska aktuálnosti informací bude pro čtenáře nezajímavá stará informace o tom, že se pohřešuje letadlo, které se ztratilo před třemi dny. Bude ho zajímat, jestli se už našlo nebo ne.

Opět by bylo možné využít zkoušecích mechanismů a dospět tak k optimálnímu rozsahu. Tentokrát se časový rozvrh určil dedukcí a odhadl na dva dny.

3.3.4 Počáteční rozklad, obecná skupina

Počáteční rozklad na obecné skupiny v seskupení není možný. S obecnou skupinou se pracuje ve smyslu, kdy nelze určit konkrétní skupinu. Zpráva, které nelze přiřadit do žádné skupiny, je charakteristická malým počtem slov.

Počáteční rozklad lze definovat jako omezení počtu potenciálních skupin, které lze využit pro hledání nejvhodnější. Tento rozklad je určen časových rozsahem jednotlivých zpráv.

3.3.5 Vzdálenost skupin

Bohužel je teoreticky i prakticky může stát, že vznikne více skupin o stejném tématu. Tyto skupiny by bylo možné následně sjednocovat, ale práce poukazuje na tuto možnost pouze jako rozšíření.

Podmínky, které vedou ke vzniku překrývajících se skupin

1. Zprávy obsahují jiné formulace, slova mohou být stejná, ale v jiném pádu nebo čase.
2. Zprávy obsahují synonyma klíčových slov
3. Zpráva, která vytvořila skupinu, byla rozsahově velice odlišná od jiné zprávy, která si vytvořila svoji vlastní skupinu.

Odlišné formulace

Tato nevýhoda by šla odstranit pomocí lemmatizátoru, který by slova o stejném základu měl v jedné skupině.

Implementace lemmatizátoru by byla velmi náročná a byla by nad rozsah této práce.

Synonyma

Slova stejně znějící, ale odlišného významu. Aby se ošetřila tato skutečnost, bylo by nutné implementovat slovník synonym.

Implementace slovníku synonym by byla velmi náročná a byla by nad rozsah této práce.

Rozdílnost v délce obsahu

Tato skutečnost je potřebná obecně pro správně zařazení článku do shluku. Pokud by skupina vznikla z dlouhého článku a obsahovala spoustu slov, může se stát, že přijde krátký článek, který nebude o stejném tématu, ale náhodou může ve svém obsahu mít nějaká základní slova z dlouhého článku. Tomuto lze předejít pokud, zprávy ve stejné skupině budou přibližně stejně rozsáhlé. Stejná délka je v praxi ovšem nereálná.

Definujme si hodnotu doporučeného poměru, který bude využívat výše určenou relevanci. Pro všechny ostatní případy zavedeme relevanci větší.

Pro hodnotu doporučeného poměru je použito označení **ratioKeywordProposed**. Testováním se dospělo k nastavení 30% prahu.

Výpočet aktuálního poměru

```
ratioKeywordActual = TRUNCATE(sumKeywords / sumKeywordsGroup,4);
```

Kde:

sumKeywordsGroup *– je součet všech slov ve skupině*

sumKeywords *– je součet všech slov zprávy*

TRUNCATE *– s parametrem 4 odtrhne desítitisíciny*

Výsledkem je poměr mezi počtem slov skupiny a zprávy. Tento poměr se porovnává s doporučeným poměrem. Je-li poměr menší, využívá se nastavení relevance na 50 procent.

3.3.6 Implementace rozpoznávání skupiny

Získání skupiny článku je opět řešené na úrovni databázového systému pomocí uložených procedur.

Články, které nově přibudou do databáze, nejdříve projdou procedurou vytvoření obsahu. Pokud mají více jak 10 slov, jsou zařazeny do procesu rozpoznávání. Mají nastavený atribut kontent na 1 a prázdné místo (NULL) pro označení skupiny.

Aby nedocházelo k velkému zpracování dat najednou, začne se tvořit rozpoznávat pouze posledních několik článku. Řazení článku, které je třeba takto rozebrat, je provedeno speciálním způsobem podle rozpoznané kategorie od poslední úrovně.

Následující dotaz vybere všechny zprávy, ve kterých se bude rozpoznávat skupina.

```
SELECT idItemRSS FROM itemrss i
WHERE idgroup IS NULL AND content=1
ORDER BY pubdate LIMIT 1000;
```

Pro každé identifikační číslo se volá funkce **recogGroupArticleById2**, která hledá nejvhodnější skupinu nebo vytváří novou

3.3.7 Tabulka grouparticle_part

Tabulka grouparticle_part v sobě ukládá skupiny RSS článků.

Sloupce tabulky

Každá skupina je označena svým číslem, datem článku, stavem a číslem kategorie. Datum článku kopíruje datum článku publikace. Skupina je aktivní, pokud je mladší než dva dny.

grouparticle_part 1

Key	Column Name	Datatype	Not Null	Default	Comment
PK	idGroup	INT(10)	Yes		Číslo skupiny
PK	articlesDate	DATETIME	Yes		Datum vytvořeni
	status	ENUM('active','archive')	Yes	'active'	Stav skupiny
	idCategory	MEDIUMINT(8)	No	NULL	Číslo kategorie

Indexy

Primární klíč je složený s atributu čísla skupiny a data článku. Ještě je zde vytvořený index nad atributem čísla kategorie, který může obsahovat duplicity. Kategorií, by se dal optimalizovat výběr potenciálních skupin, do kterých se může článek zařadit. Tím je bohužel spojená téměř 100% funkčnost správného zařazení. Takže je to motivace na rozšíření aplikace.

grouparticle_part 2

Index Name	Columns	Primary	Unique	Type
PRIMARY	idGroup, articlesDate	Yes	No	PRIMARY
idCategory	idCategory	No	No	INDEX

Partition

Tabulka využívá vlastnosti partitioningu, který je vytvořený z rozsahu hodnot. Rozsah definuje funkce TO_DAYS, která vrací číslo dne od začátku letopočtu. Díky ní, lze vytvořit partitiony třeba po 5, 10, 20 dnech.

```
PARTITION BY RANGE(TO_DAYS(articlesDate)
```

Poznámka

Ve výsledku se partitioningu využívat nemusí, protože se pracuje s pouze s aktivními skupinami. Bylo by možně využít tabulku pro odlévání nepotřebných dat, podobně jako to využívají archivační tabulky pro obsah článku nebo skupin.

Vztahy (cizí klíče)

Pokud tabulka využívá partitioningu, není možné nad ní definovat cizí klíč. Pouze zpětně se na ní mohou odkazovat jiné tabulky. Na tabulku grouparticle_part se odkazuje grouparticlecontentnew.

grouparticle_part 3

Relationship Name	Parent Table	Child Table	Card.
FK_grouparticlecontentnew_grouparticle	grouparticle_part	grouparticlecontentnew	1:n

3.3.8 Tabulka grouparticlecontentnew

Tabulka grouparticlecontentnew v sobě ukládá obsah skupiny. Obsah vychází z klíčových slov článku, který skupinu vytvořil a má tedy 100% relevanci.

Sloupce tabulky

Nejdůležitějšími atributy jsou číslo skupiny a číslo klíčového slova. Dále se v tabulce nachází počet výskytů slova ve skupině.

grouparticlecontentnew 1

Key	Column Name	Datatype	Not Null	Comment
PFK	idGroup	INT(10)	Yes	Číslo skupiny
PFK	idKeyword	INT(10)	Yes	Číslo klíčového slova
	countKey	SMALLINT(5)	Yes	Počet výskytů slova

Indexy

Primární klíč je složený z čísla skupiny a čísla článku, protože v jedné skupině je více slov a stejné tak stejné slovo může být v různých skupinách.

Rychlý přístup pomocí indexu využívají atributy čísla skupiny a čísla slova. V této tabulce se při rozpoznávání skupiny vyhledává podle klíčových slov. Tento index je jedním z nejdůležitějších.

grouparticlecontentnew 2

Index Name	Columns	Primary	Unique	Type
PRIMARY	idGroup, idKeyword	Yes	No	PRIMARY
keyword	idKeyword	No	No	INDEX
idgroup	idGroup	No	No	INDEX

Vztahy (cizí klíče)

Tabulka se odkazuje pomocí cizího klíče do tabulek grouparticle_part a keywordsall.

grouparticlecontentnew 3

Relationship Name	Parent Table	Child Table	Card.
FK_grouparticlecontentnew_grouparticle	grouparticle_part	grouparticlecontentnew	1:n
FK_grouparticlecontentnew_keywordsall	keywordsall	grouparticlecontentnew	1:n

3.3.9 Procedura recogGroupArticleById2

Procedura, která se snaží nalézt nejvhodnější skupinu pro článek a pokud není nalezena, nebo nesplňuje příslušné požadavky, vytvoří novou skupinu.

Uložená procedura implementuje následující algoritmus:

1. Vezmi z článku datum vydání a číslo kategorie.
2. Zjisti počet slov z tabulky s obsahem.
3. Vyhledej skupinu s největším počtem slov z rozebíraného článku.
4. POKUD byla nalezena skupina, zjisti:
    a. Počet slov v článku
    b. Počet slov ve skupině článku
    c. Kategorii skupiny
5. Zjisti aktuální relevanci a poměr v rozsahu článku.

6. POKUD je poměr mezi rozsahy článků **větší** než minimální doporučený a současně je aktuální relevance větší než doporučená NEBO je poměr mezi rozsahy článků **menší** než minimální doporučený a současně je aktuální relevance větší než doporučená relevance pro menší poměr, byla nalezena vhodná skupina.
7. POKUD je nalezena skupina, přiřaď ji k článku a zkontroluj kategorie.
8. Je-li kategorie článku vetší než kategorie skupiny, přiřaď kategorii ke skupině a aktualizuj v ní všechny zprávy.
9. POKUD nebyla nalezena skupina, vytvoř novou skupinu, číslo nové skupiny přiřaď k článku a vytvoř obsah skupiny.

Vyhledání skupiny

Nejvhodnější skupina využívá dotazu se seskupením podle kategorie a počítá součet nalezených slov.

```
SELECT gc.idGroup ,
  SUM(IF(gc.countkey> a.countkey,a.countkey,gc.countkey)) as pocet
  INTO idGroup1, sumFoundKeywords
FROM grouparticlecontentnew gc
  LEFT JOIN articlecontentnew a on gc.idkeyword = a.idkeyword
  JOIN grouparticle_part g on gc.idGroup = g.idGroup
WHERE a.idItemRSS = id AND status = 1
  GROUP BY gc.idGroup ORDER BY pocet desc LIMIT 1;
```

3.4 Získání kategorie

Další typ seskupení je vytvořený podle kategorie.

Struktura kategorií je uložena v tabulce s rekurzivní relací. Tato tabulka byla konstruována ručně. RSS zprávy sice mohou obsahovat element category, ale takto pořízenou informaci nelze zařadit na správné úrovně.

Automatické vytváření kategorií by bylo možné jen pouze pro jednu úroveň. Jediná teoretická eventualita, jak je vytvořit bez lidské síly, je rozebrat strukturu webu, který je má dostatečné propracované.

Kategorie, které se ručně identifikovali, nemusí být dostačující. Jde o slova, kterým může daná kategorie charakterizovaná. Například kategorie Zprávy, může být na jiných zdrojích Tisk a pre-press, Tisková řešení, Událo se, Události, Zpráva. S touto skutečností je potřeba ještě další tabulka, která v sobě uchová relevantní názvy kategorií.

3.4.1 Tabulka category

Tabulka category v sobě ukládá názvy kategorií a jejich úroveň.

Sloupce tabulky

Sloupec idMainCategory v sobě nese informaci o kategorii v nadřazené úrovní (rodičovské kategorii).

category 1

Key	Column Name	Datatype	Not Null	Default	Comment
PK	idCategory	MEDIUMINT(8)	Yes		Číslo kategorie
	name	CHAR(30)	Yes		Název
	idMainCategory	MEDIUMINT(8)	No	NULL	Číslo nadřazené kategorie

Indexy

Primární klíč je definovaný nad číslem kategorie, protože tato čísla budou jedinečná. Nad číslem nadřazené kategorie je definovaný index při rychlejší přístup a především pro vytvoření referenční integrity.

category 2

Index Name	Columns	Primary	Unique	Type
PRIMARY	idCategory	Yes	No	PRIMARY
idMainCategory	idMainCategory	No	No	INDEX

Vztahy (cizí klíče)

Tabulka category je sama v sobě v relaci, využívá rekurzivní relaci a referenční integritu. Nelze se odkazovat na kategorii, která neexistuje.

Na tabulku category se ještě odkazují další tabulky. Jde o categoryrelevantname, která v sobě nese relevantní názvy kategorií.

category 3

Relationship Name	Parent Table	Child Table	Card.
FK_category_category	category	category	1:n
FK_categoryrelevant_category	category	categoryrelevant	1:n

3.4.2 Tabulka categoryrelevant

Sloupce tabulky

Relevantní název může být složený z více slov nebo části slova. Rozpoznávání probíhá pomocí operátoru LIKE. Pokud je možné, že se slovo může být součástí jiného slova, lze doplnit mezery. Dalo by se pracovat s regulárními výrazy, ale je to motivace spíše na rozšíření práce, protože by se musela optimalizovat výkonnost.

categoryrelevant 1

Key	Column Name	Datatype	Not Null	Default	Comment
PFK	idCategory	MEDIUMINT(8)	Yes	'0'	Číslo kategorie
PK	relevantName	CHAR(80)	Yes		Relevantní název

Indexy

Primární klíč je kombinovaný a skládá se s atributů číslo kategorie a relevantního názvu. V tabulce se bude dost často vyhledávat pouze podle čísla kategorie. Z tohoto důvodu je nad sloupcem definován index.

categoryrelevant 2

Index Name	Columns	Primary	Unique	Type
PRIMARY	idCategory, relevantName	Yes	No	PRIMARY
idCategory	idCategory	No	No	INDEX

Vztahy (cizí klíče)

Provázání s tabulkou category je vytvořeno pomocí referenční integrity, kde je důležité vyřešit zachování referenční integrity. Při odstranění i aktualizaci je použito omezení CASCADE.

categoryrelevant 3

Relationship Name	Parent Table	Child Table	Card.
FK_categoryrelevant_category	category	categoryrelevant	1:n

3.4.3 Rozpoznávání kategorie

Rozpoznání kategorie zprávy je možné provést ze čtyř základních zdrojů.

Kategorie RSS zprávy

První možností je využití struktury RSS feedu. Standart RSS 2.0 definuje element category, kde lze uvést kategorii.

Výhody

- Snadno lze zjistit kategorii pouze přečtením hodnoty elementu.
- Lze využít pro potenciální tvorbu nových kategorií.

Nevýhody

- Element category není k dispozici ve starších standardech.
- Kategorie může být obecná.
- V názvu kategorie mohou být navíc další slova, která nesouvisí s kategorií, například označení zdroje.

Titulek RSS zprávy

Druhou možností je rozbor titulku zprávy. Je velice pravděpodobné, že v titulku bude název nebo relevantní název kategorie.

Výhody

- Titulek obsahuje téměř každá zpráva.
- Slova titulku umožní nalézt pravděpodobně tu nejvhodnější kategorii.

Nevýhody

- Konkrétní slovo v titulku nemusí být obsaženo v kategoriích nebo relevantních názvech.
- Nelze využít celý titulek pro nové kategorie.

Link RSS zprávy

Další možností je rozbor internetového odkazu zprávy, protože je možné využít SEO optimalizace zdrojů pro získání informace už jen z odkazů.

Výhody

- Link obsahuje každá zpráva.
- Při propracované SEO optimalizaci lze nalézt odpovídající kategorii.

Nevýhody

- Všechny linky na články mohou odlišné pouze identifikátorem zprávy.
- Linky neobsahují speciální znaky, pokud je v názvu kategorie nebo relevantním názvu mezera, nelze jej využít pro odkazy.

Kategorie RSS zdroje

Poslední možností je využití RSS zdroje. Některé RSS už jsou uspořádané do kategorií a lze tuto kategorii rozpoznat podobnými způsoby jako u samotných zpráv.

Výhody

- Pokud se nerozpozná kategorie ze zprávy, je možné využít RSS zdroj.

Nevýhody

- Kategorie nemusí být pro RSS zdroj definovaná.
- Kategorie může být obecná.

3.4.4 Implementace rozpoznání

Výsledná implementace využívá uspořádání kategorií, kdy poslední úroveň má nejvyšší číslo.

Rozpoznávání kategorie probíhá bezprostředně po načtení RSS zprávy.

Samotné hledání vhodné kategorie využívá jak tabulku kategorie, tak tabulku relevantních názvů.

Rozpoznání má na starost funkce getIDCategoryByName, která má jeden parametr, a to slovo čí fráze, která se použije pro hledání vhodné kategorie.

Funkce getIDCategoryByName

Implementuje následující algoritmus:

1. Vrať nejvyšší číslo kategorie, kde se rovná parametr názvu kategorie nebo relevantnímu názvu.
2. Pokud je číslo kategorie menší než jedna, vrať nejvyšší číslo kategorie, kde parametr je obsažen v názvu kategorie nebo relevantním názvu.
3. Pokud je číslo kategorie menší než jedna, vrať nejvyšší číslo kategorie, kde název kategorie nebo relevantním název je obsažen v parametru.

Kompletní algoritmus:

Celý postup, který se využije pro získání kategorie je následující:

1. Pokud existuje element category, zjisti číslo kategorie pomocí funkce getIDCategoryByName, jako parametr použij hodnotu elementu, výsledek ulož do pole.
2. Zjisti číslo kategorie pomocí funkce getIDCategoryByName, jako parametr použij titulek zprávy, výsledek ulož do pole.
3. Zjisti číslo kategorie pomocí funkce getIDCategoryByName, jako parametr použij link zprávy, výsledek ulož do pole.
4. Ulož do pole číslo kategorie RSS zdroje.
5. Vrať maximum z pomocného pole a přiřaď k vlastnosti idCategory u RSS zprávy.

3.5 Řazení skupin RSS zpráv

Posledním důležitým faktorem úspěšnosti poskytování informací je řazení zpráv. Pro čtenáře je jistě zajímavé, když vidí hlavní události dne mezi prvními.

Faktory, které ovlivňuje pořadí, jsou stáří nejnovější zprávy a počet zpráv ve skupině.

Obecně lze hodnotit podle následujících údajů:

- Čím více zpravodajských portálů informaci publikovalo, tím jde o důležitější skupinu zpráv.
- Čím čerstvější je nejnovější zpráva ve skupině, tím se jedná o aktuálnější skupinu zpráv.

Obě hlediska je třeba brát v úvahu. Koeficient řazení počítá funkce countOrder, která parametrem dostane počet zpráv ve skupině a datum vydání zprávy.

3.5.1 Funkce countOrder

Funkce implementuje následující algoritmus:

1. Pokud je čas publikace větší menší než aktuální, pokračuj dál, jinak nastav výsledek na nulu.
2. Vypočítej rozdíl obou časů v minutách.
3. Je-li rozdíl menší než hodina, zvyš prioritu řazení, za každých 20 minut o 1.
4. Je-li rozdíl větší než hodina, vypočítej rozdíl obou časů v hodinách.
5. Sniž prioritu řazení za každé dvě hodiny o 1.
6. Pokud je potřeba priorita snížit prioritu více než je počet zpráv ve skupině, nastav výsledek na 0, jinak vrať správný výsledek.

4 Vývoj aplikace

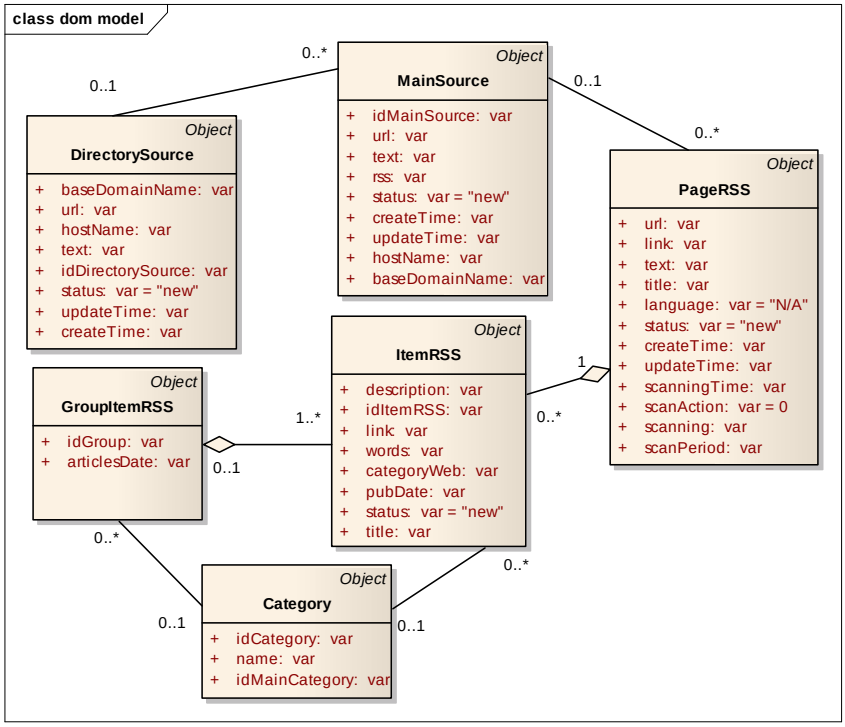
4.1 Analýza

Vytvořené řešení bude zdroje zpráv získávat co nejvíce automaticky. Ze zdrojů je třeba identifikovat stránky, na kterých se nalézají odkazy na novinky. Předpokládá se, že každá aktualita bude na samostatné stránce s pevným url. Dále je třeba získat obsah zprávy, aby bylo možné ve zprávách hledat a seskupovat je podle stejného tématu. Je nutné zachytit všechny publikované zprávy a provádět skenování podle časové potřeby.

4.1.1 Business analýza

Základní business objekty

- Katalogový zdroj – Odkaz na stránku z internetového katalogu, který obsahuje odkazy na zpravodajské servery.
- Hlavní zdroj – Odkaz rozpoznaný z katalogu, který obsahuje novinky, články, zprávy, recenze apod.
- RSS zdroj – Veškeré publikované novinky se budou čerpat z RSS zdrojů. Získání ze samotné stránky je součástí motivačních vylepšení aplikace.
- RSS zpráva – Zpráva získaná z RSS zdroje.
- Skupina zpráv – Skupina zpráv se stejným obsahem.
- Kategorie – Kategorie pro RSS zdroj, zprávu nebo skupinu.



Doménový model 1

4.1.2 Analýza požadavků

Funkční požadavky

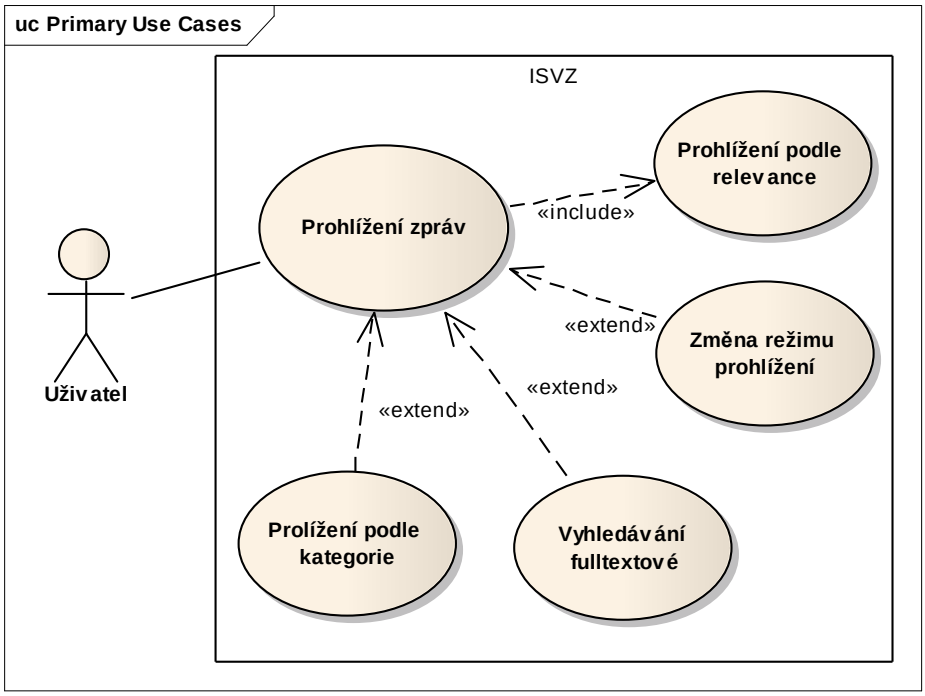
Systémové požadavky

- Systém rozpozná hlavní zdroje.
- Systém rozpozná RSS zdroje.
- Systém bude pravidelné skenovat RSS zdroje a získávat RSS zprávy.
- Systém bude vytvářet svůj vlastní slovník klíčových slov.
- Systém bude pravidelně vytvářet obsah nových zpráv.
- Systém bude pravidelně vytvářet skupiny zpráv.
- Systém bude publikovat zprávy v základní seskupené podobě.
- Systém bude rozpoznávat kategorie článků.

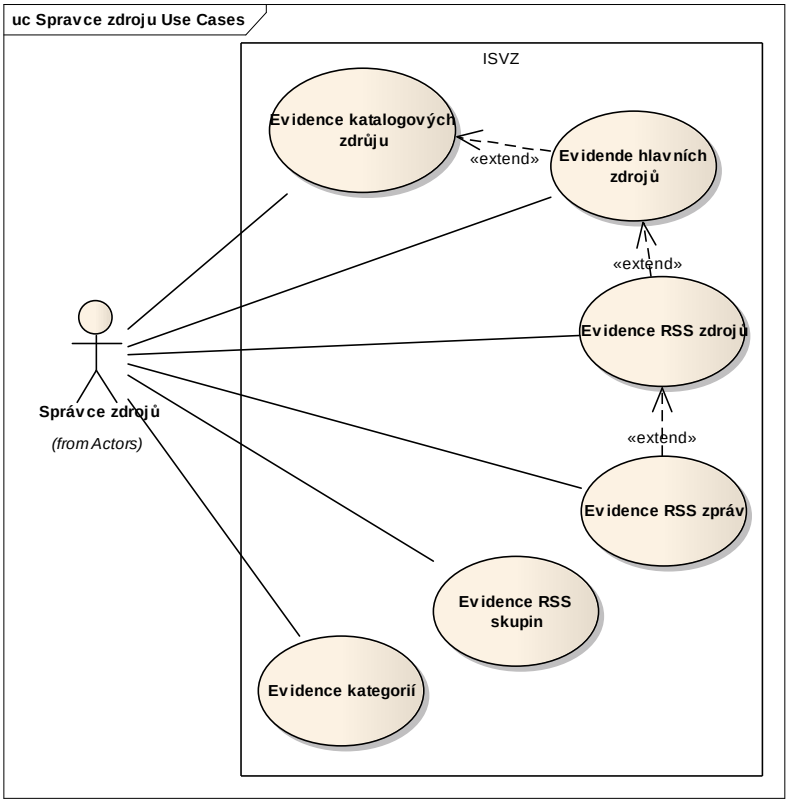
Uživatelské požadavky

S aplikací budou pracovat dva typy uživatelů. Obecný anonymní uživatel, který bude moci prohlížet novinky a správce zdrojů, který bude moci administrovat všechny objekty.

Případy užití pro uživatele ukazuje obrázek Use Case Diagram 1. Případy užití pro správce zdrojů ukazuje obrázek Use Case Diagram 2.



Use Case Diagram 1



Use Case Diagram 2

Rozpoznání hlavního zdroje

Při rozpoznávání hlavního zdroje lze využít následujících skutečností:

- Všechny hlavní zdroje z každého katalogu musí mít absolutní adresu.
- Všechny odkazy budou jiný název domény.
- Odkazy, které ukazují hlavní zdroj, mají stejnou strukturu a formátování.
- Elementy s odkazy budou ve struktuře stránky poslední úrovni.

Hledání RSS zdroje

Při hledání RSS zdroje lze využít následujících skutečností:

- Odkaz na RSS zdroj může být umístěný v elementu link.
  - `<link rel="alternate" type="application/rss+xml" href=" " title=" " />`
- Odkaz bude obsahovat slovo rss v url.
- Server může poskytovat více RSS feedů strukturovaných do kategorií.
- RSS má svoji danou strukturu, obsahuje elementy channel, link apod.

Získání RSS zpráv

Při získání RSS zpráv lze využít následujících skutečností:

- RSS zdroj obsahu RSS zpráv, kde každá zpráva může obsahovat popisy
  - title – titulek zprávy
  - link – pevný odkaz na zprávu
  - description – krátký popis zprávy
  - pubDate – datum zveřejnění
  - category – název kategorie
- description poslouží pro práci s obsahem zpráv, pokud není k dispozici, nelze provést seskupení podle obsahu
- pubDate bude použit pro datum zprávy, pokud není k dispozici, nebo obsahuje špatný formát, nebo je špatně určený, lze doplnit aktuální datum a čas
- category je možné použít pro přiřazení kategorie

Získání skupiny RSS zpráv a kategorie

Pro získání skupiny lze využít shlukovou analýzu. Podrobněji bude řešeno v kapitole o dobývání informací.

Nefunkční požadavky

Transport (Přenos)

- Požadavky z klienta obslouží http protokol.

Persistence (Uložení)

- Data budou uložena v relačním DS s podporou integrity a partitioning.

Security(Zabezpečení)

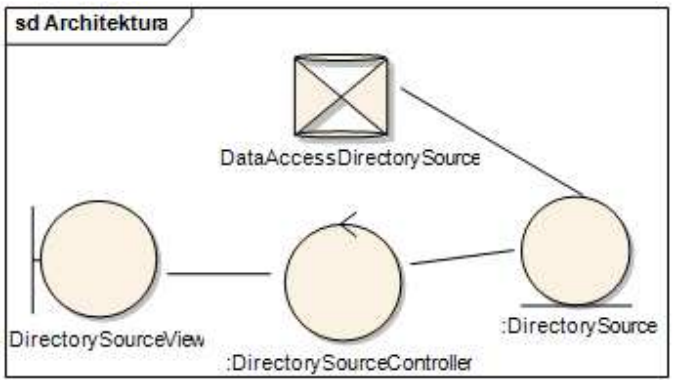
- K prohlížení může přistupovat kdokoliv, administrace podléhá pouze přístupovým právům.

Performance (Výkonnost)

- Vrácení výsledku po hledání nebo načtení stránky musí mít maximální odezvu v řádech sekund.

4.2 Architektura a design

Aplikace využívá architekturu MVC(Model, View, Controller) +P (perzistence databázová vrstva). Pro každý doménový objekt je samostatný MVC+P. Komunikační diagram je na obrázku Architektura 1.

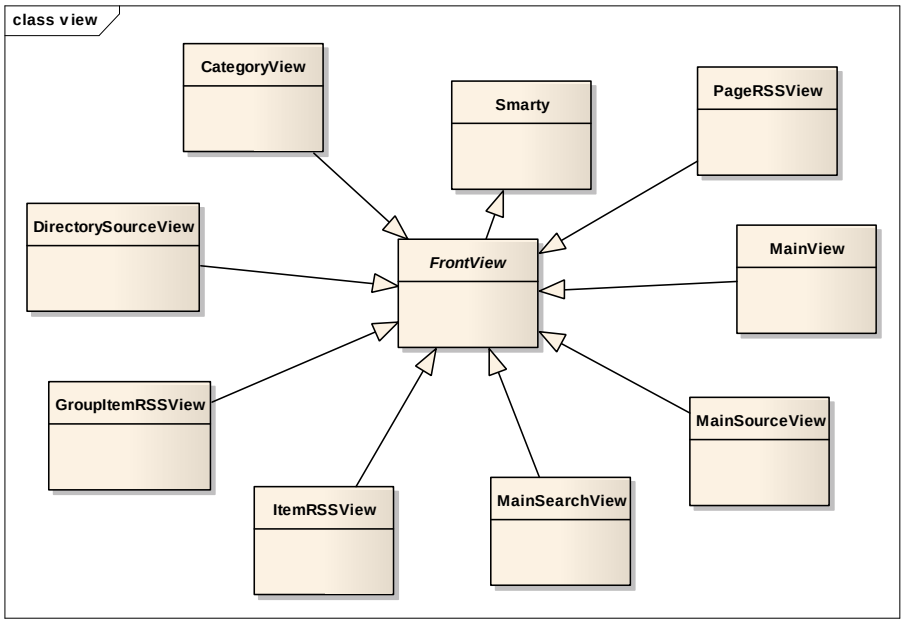


Architektura 1

4.2.1 Prezenční vrstva

Prezenční vrstva vytváří výslednou podobu celé aplikace. V souvislosti s webovými aplikacemi se jedná o generování HTML kódu. ISVZ využívá šablonovací systém Smarty, ze kterého odvozuje třídy pohledu pro každý objekt.

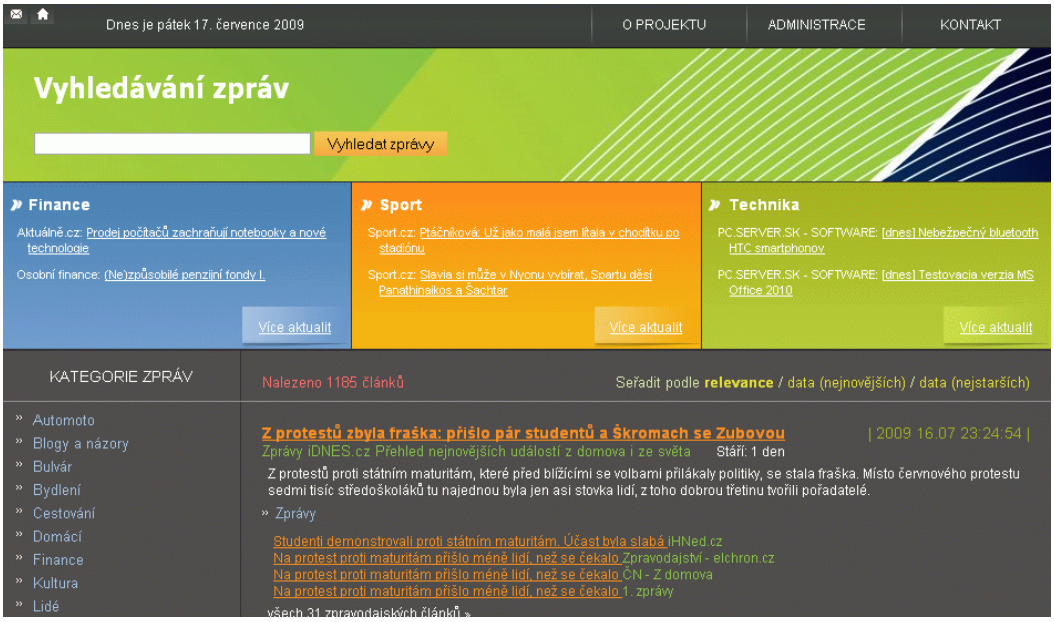
Od třídy Smart je odvozena abstraktní třída FronView, která ošetřuje základní GET a POST parametry. Tyto parametry mohou být obecné, a tudíž je může zpracovat obecný konstruktor. Celkový pohled nabízí obrázek Class diagram 1.



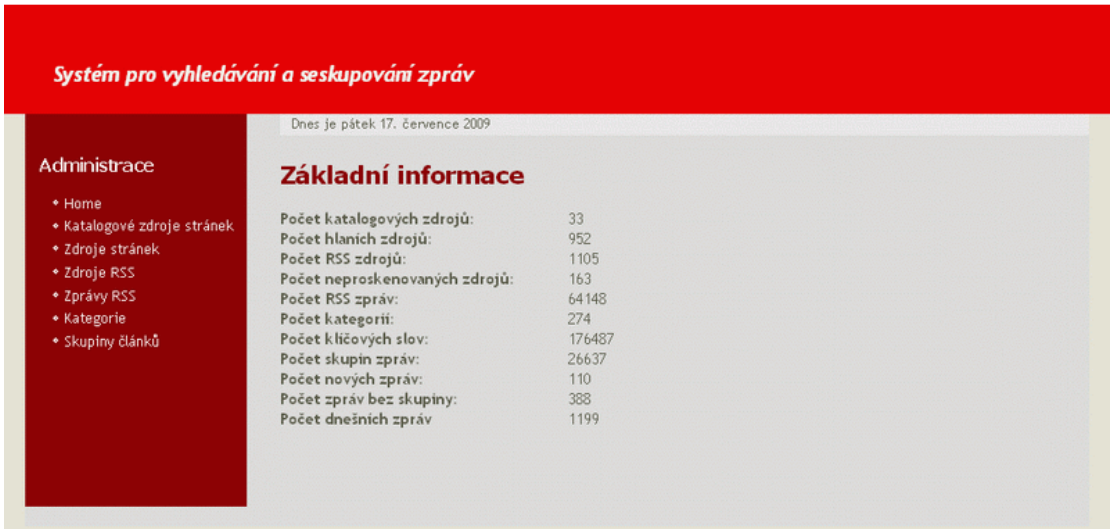
Class Diagram 1

Šablony jsou postaveny na dvou HTML layoutech. Jeden tvoří administrační rozhraní a druhý tvoří uživatelský pohled na zprávy.

Na obrázku Šablona 1 je uživatelský pohled na obrázku Šablona 2 administrační.



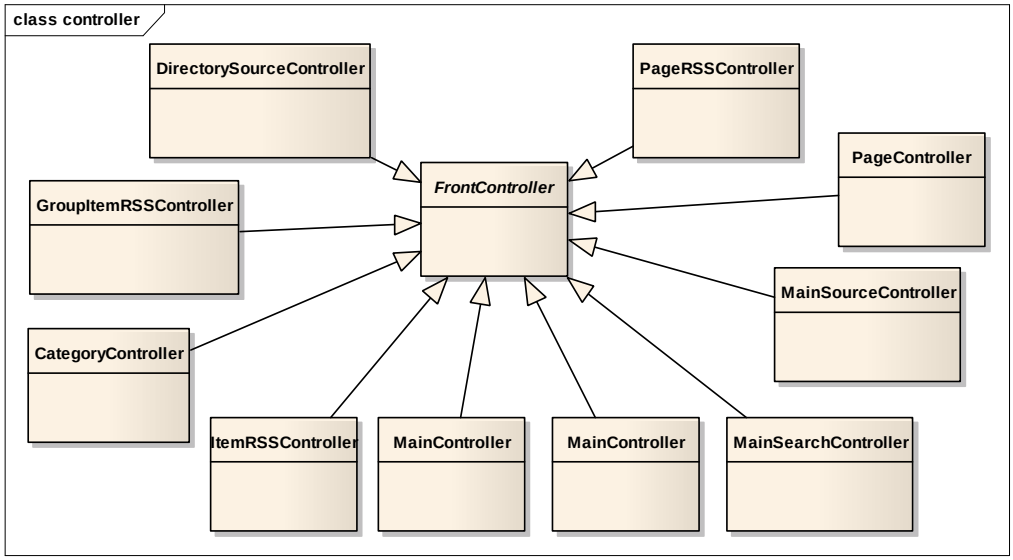
Šablona 1



Šablona 2

4.2.2 Kontrolní vrstva

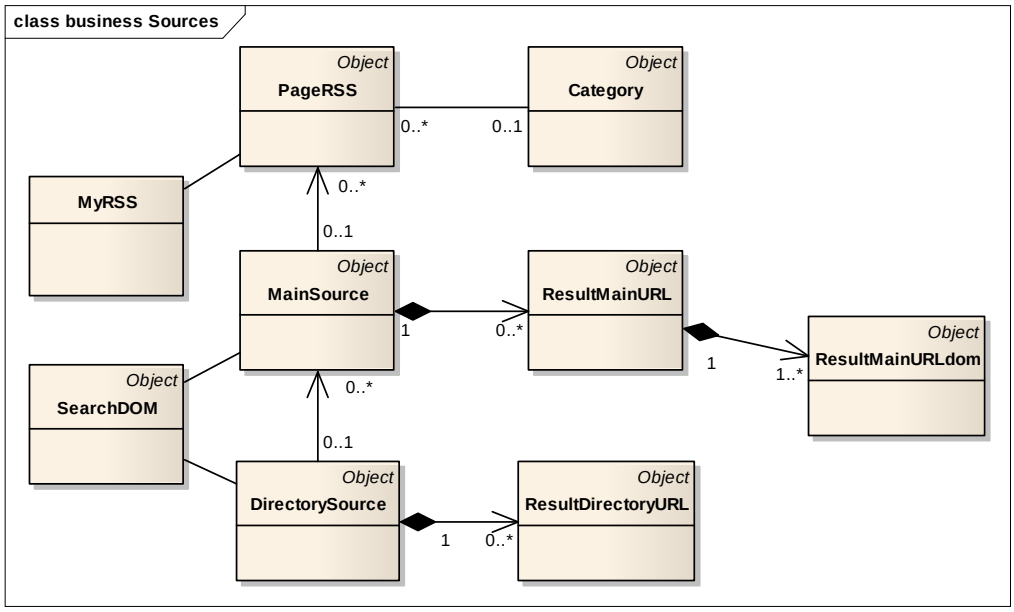
Kontrolní vrstva získává data z prezenční vrstvy a zpracovává logické business vrstvě. ISVZ zachovává stejný přístup jako v prezenční vrstvě. Základní pohled na třídy je k dispozici na obrázku Class Diagram 2.



Class Diagram 2

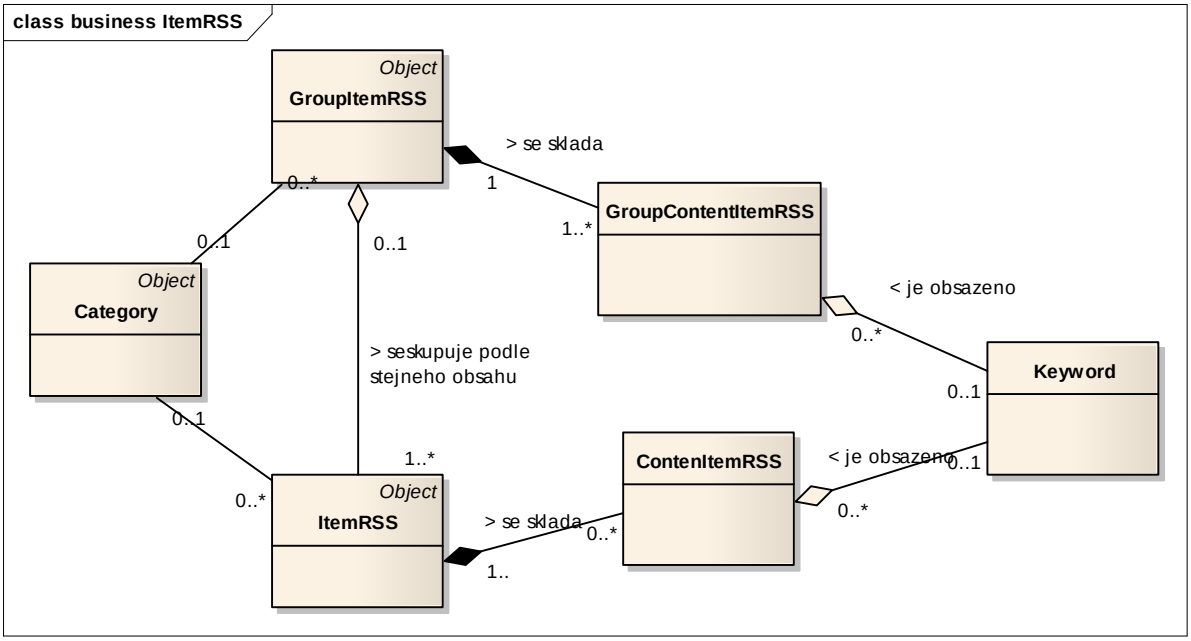
4.2.3 Logická vrstva

Logická vrstva řeší samotnou logiku aplikace. Pohled je rozdělen na dva diagramy. Na obrázku Class Diagram 3 je znázorněn pohled na zdroje.



Class Diagram 3

Na obrázku Class Diagram 4 je znázorněn pohled na zprávy, skupiny zpráv a klíčová slova.

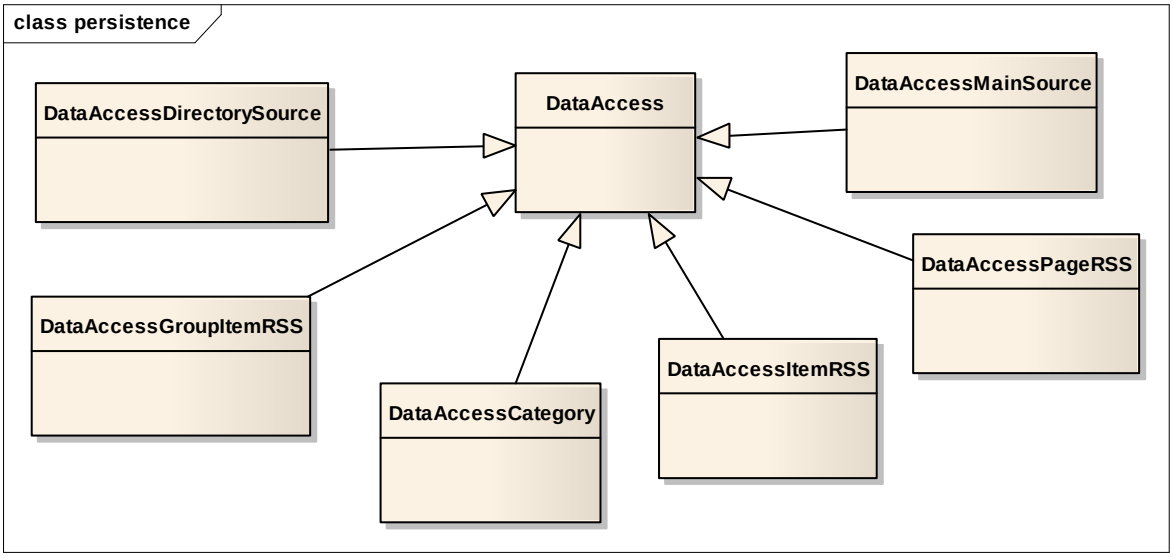


Class Diagram 4

4.2.4 Databázová vrstva

Každý doménový objekt má svoje třídy, které zajišťují přístup k databázovým tabulkám.

Pohled na třídy je na obrázku Class Diagram 5.



Class Diagram 5

4.3 Databázový systém

Pro uložení persistentních dat byla zvolena databáze MySQL, verze 5.1. Tento relační databázový systém je vhodný nejrůznější webové aplikace, které potřebují jak snadnou implementaci, tak vysoký výkon. Již od začátku bylo jasné, že bude třeba pracovat s velkým objemem dat a bude nutné využit co nejrychlejšího přístupu pomocí indexování nebo partitioningu.

Univerzálnosti řešení lze dosáhnout použitím uložených procedur a funkcí.

Tabulky

Seznam tabulek naleznete v příloze v tabulce Databázové objekty 1.

Procedury s funkce

Seznam uložených procedur a funkcí naleznete v příloze v tabulce Databázové objekty 2.

Události

Seznam událostí naleznete v příloze v tabulce Databázové objekty 3.

Struktura databáze

Obrázek s kompletním diagramem databáze je umístěn v příloze na obrázku ER diagram struktury databáze 1.

5 Výsledky seskupení

Na následujícím obrázku je pohled na skupinu a relevanci v administračním prostředí.

Na následujícím obrázku je pohled na skupinu v uživatelském prostředí.

6 Automatické zpracování

Pokud má aplikace fungovat bez lidského faktoru, musí sama pravidelně provádět skenování, vytváření obsahu zpráv a rozpoznávání skupin.

6.1 Automatické zpracování na úrovni aplikace

Na úrovni aplikace je nutné spouštět skript, který bude skenovat zdroje, kterým už uběhla perioda skenování.

Pro tyto potřeby má operační systém Windows naplánované úlohy a operační systém Linux crontab.

6.2 Automatické zpracování na úrovni databázového systému

Databázový systém MySQL umožňuje plánování událostí přímo definicí pomocí SQL dotazu. Pracuje s tzv. EVENTy, které se mohou vykonat jednou nebo pravidelně v nastaveném intervalu. K plánování událostí je třeba puštěného deamona. Jde o nastavení globální proměnné MySQL event_scheduler. Výchozí nastavení je OFF, administrační rozhraní umožňuje pohodlnou změnu.

Události, které jsou naplánované, se nacházejí v příloze v tabulce Databázové objekty 3.

7 Problémy v průběhu řešení

7.1 Výpadky serverů

Velmi častým problémem může být výpadek serverů, který způsobí zpomalení stahování nových zpráv. Skript by se stále snažil přistoupit ke stránce a stáhnout její obsah, přitom server může být dočasně nedostupný.

Toto může vyřešit rozšíření PHP o curl, který definuje timeout pro snahu o připojení k serveru.

7.2 Zjišťování periody zpracování

Servery, které poskytují nové informace, mají různě dlouhé intervaly mezi vydáváním zpráv. Některé publikují nové informace téměř každou minutu, jiné jednou za den několik práv. Je zbytečné skenovat nějaký zdroj každou minutu, když vytváří jednu zprávu za den. K této optimalizaci poslouží tabulka, která zaznamenává každé skenování.

7.2.1 Tabulka scanningpagerss

Tabulka monitoruje skenování jednotlivých RSS zdrojů a ukládá počty nově naskenovaných zpráv.

Sloupce tabulky

Tabulka obsahuje číslo RSS zdroje, datum a čas skenování, datum a čas předchozího skenování, datum poslední novinky a počet nových zpráv. Všechny sloupce jsou potřebné pro určení nové periody.

Tabulka v sobě ukládá pouze skenování za poslední den, každý den se přelévají data do archivační tabulky, která obsahuje stejné atributy.

scanningpagerss 1

Key	Column Name	Datatype	Not Null	Default	Comment
PFK	idPageRSS	INT(10)	Yes	'0'	Číslo RSS zdroje
PK	scanningTime	TIMESTAMP	Yes	CURRENT_TIMESTAMP	Čas skenování
	scanningFromTime	TIMESTAMP	Yes	'0000-00-00 00:00:00'	Čas předchozího skenovaní
	lastNewsTime	TIMESTAMP	Yes	'0000-00-00 00:00:00'	Datum poslední zprávy
	countNews	SMALLINT(5)	Yes		Počet nových zpráv

Indexy

Primární klíč je složený z čísla RSS zdroje a data a času skenování.

scanningpagerss 2

Index Name	Columns	Primary	Unique	Type
PRIMARY	idPageRSS, scanningTime	Yes	No	PRIMARY

Vztahy (cizí klíče)

Číslo RSS zdroje je svázáno referenční integritou, která nastavuje kaskádové aktualizace i odstranění.

scanningpagerss 3

Relationship Name	Parent Table	Child Table	Card.
FK_scanningpagerss_pageRSS	pagerss	scanningpagerss	1:n

7.2.2 procedura newScanPeriod

Procedura vypočítává novou periodu skenování a se spouští jednou za den.

Algoritmus pro vytvoření nové periody

1. Pro každé skenování urči:
    a. interval přidání jednoho nového článku
    b. dobu od vydání posledního článku ke skenování
2. Pro každý zdroj urči:
    a. průměrný interval přidání jednoho nového článku
    b. průměrnou dobu od vydání posledního článku ke skenování
3. Porovnej interval a dobu vydání mezi sebou.
4. Nový interval je na menší hodnota.
5. Proveď aktualizace pro skenované zdroje.
6. Přesun všechna skenování do archivační tabulky.
7. Vyprázdní tabulky skenování.

8 Jiná řešení

Google news

Google news je jedna s jejich významných služeb společnosti Google. Výsledkem je publikace nejnovějších informací z tisíce informačních zdrojů z celého světa na jednou místě.

(1)Služba Zprávy Google shromažďuje události z více než 400 zdrojů zpráv v českém jazyce a automaticky je seřazuje tak, aby nejdůležitější zprávy byly prezentovány jako první. Témata jsou aktualizována každých 15 minut.

K zobrazování zpráv je možné zvolit několik režimů prohlížení. Nejzajímavějším je seskupení podle obsahu (relevance). Výsledkem je skupina aktualit zaměřená na stejné téma. V této jedné skupině se nalézají novinky z různých zdrojů. Čtenář vidí hlavní událost pouze jednou a nemusí číst stejné zprávy víckrát. Nebo naopak, pokud ho problematika zaujala, může si přečíst informace z různých pramenů a vytvořit si tak kompletní obraz o celé probírané situaci.

1. zprávy

Server 1-zprávy poskytuje podobné informace jako Google news. Agreguje RSS zdroje ze zpravodajských serverů a publikuje z nich nejnovější zprávy. Přidání vlastního zdroje je pouze na vyžádání. Zprávy nejsou seskupené podle stejného tématu.

Aktuality.sms.cz

Web monitoruje 1193 zpravodajských serverů. Umožňuje vyhledávání a jednoduché kategorie.

8.2 Srovnání řešení

Google news

Výhody

- Seskupení podle tématu.
- Spousta informačních zdrojů.
- Obrázky u zpráv.
- Odeslání zprávy e-mailem.
- Řazení zpráv podle aktuálnosti a počtu zveřejnění na různých serverech.

Nevýhody

- Nelze přidat vlastní zdroj.
- Malý počet kategorií.

1-zprávy.cz

Výhody

- Vetší počet kategorií.
- Tagy pro různá témata.

Nevýhody

- Menší počet zdrojů
- Přidání na požádání a pouze v případě časté publikace nových článků.
- Neseskupuje podle stejného tématu, definuje pouze podobné stránky.
- Nepracuje s obrázky.
- Neuvádí zdroj u aktualit.

- Nelze odeslat zprávy emailem.
- Neobsahuje řazení zpráv podle aktuálnosti a počtu zveřejnění na různých serverech.

Aktuality.sms.cz

Výhody

- Po registraci lze upravit profil oblíbených zdrojů.
- Pracuje s obrázky.
- Fulltextové vyhledávání podle relevance slov.

Nevýhody

- Nedefinuje možnosti přidání zdrojů.
- Neseskupuje podle stejného tématu.
- Nelze odeslat zprávy emailem.
- Malý počet kategorií.
- Neobsahuje řazení zpráv podle aktuálnosti a počtu zveřejnění na různých serverech.

ISVZ

Výhody

- Detailně propracované kategorie do 4 úrovní.
- Rychlé získání zdrojů z katalogů.
- Žádné omezení pro potencionální zdroje.
- Seskupení podle tématu.
- Prohlížení dle kategorie.
- Fulltextové vyhledávání podle relevance slov.
- Řazení zpráv podle aktuálnosti a počtu zveřejnění na různých serverech.

Nevýhody

- Nepracuje s obrázky.
- Nelze odeslat zprávy emailem.

9 Závěr

Složitost celého problému neumožňovala dotáhnout výsledné řešení k dokonalosti, ale nabízí spoustu možností na rozšíření aplikace.

Možnosti rozšíření:

- Hlubší využití teorie a aplikace chytřejších algoritmů.
- Určení kategorie z regulárního výrazu.
- Vytvoření uživatelského rozhraní pro prohlížení zpráv.
- Možnost rychlého přidání vlastního RSS zdroje.
- Vytvoření administračního rozhraní pro publikaci vlastních zdrojů.
- Rozložení výkonu do více uzlů a optimalizace výkonnosti.
- Implementace lemmatizátoru.
- Implementace slovníku synonym.
- Rozšíření pro více jazyků.
- Implementace obrázku ke zprávám.
- Odesílání zprávy emailem.
- Získávání zpráv ze stránky, ne pouze z RSS zdroje.

Celé řešení je ovšem naprosto funkční a poskytuje informace v požadované formě. Výsledkem je ucelený funkční program s možností dalšího rozvoje. Po implementaci výše jmenovaných rozšíření je možná i konkurenceschopnost s největším gigantem internetu jako je Google.

Řešení bakalářské práce pro mne bylo velice přínosné a přineslo mi spoustu nových cenných poznatků a informací.

10 Použitá literatura a informační zdroje

1. Zprávy Google. *Zprávy Google.* [Online] [Citace: 27. Červen 2009.] http://news.google.cz/intl/cs_cz/about_google_news.html.

2. **Žák, L.** Shluková analýza. *Automatizace.* 2004.

3. **Řezanková, H., Húsek, D. a Snášel, V.** *Shluková analýza dat, 2. vydání.* Praha : Professional Publishing, 2009. 978-80-86946-81-8.

4. 1. Zprávy - O webu. *1. Zprávy.* [Online] [Citace: 30. Červen 2009.] http://www.1-zpravy.cz/about/.

5. **Kelbel, J. a Šilhán, D.** *Shluková analýza.* 2003.

6. **Axmark, D. a Widenius, M.** *MySQL Reference Manual.* 2008.

7. **Pavel, B.** *Survey Of Clustering Data Mining Techniques.* 2002.

8. **Murty, M. N., Jain, A. K. a Flynn, P. J.** *Data Clustering: A Review.* 1999.

9. **Achour, M., a další.** *PHP Manual.* místo neznámé : PHP Documentation Group, 2009.

10. **Řezanková, H.** *Klasifikace pomocí shlukové analýzy.* Praha : autor neznámý, 2003.

11 Přílohy

Tabulky databázových objektů:

Databázové objekty 1

Tabulka	Popis
articlecontentarchive	Tabulka pro všechna klíčová slova článku
articlecontentnew	Tabulka pro klíčová slova nových článku
category	Tabulka pro kategorie
categoryprepare	Tabulka pro nové kategorie nebo relevantní názvy
categoryrelevant	Tabulka pro relevantní jména kategorie
curlerror	Tabulka pro chyby při skenování zpráv
directorysource	Tabulka pro katalogové zdroje
grouparticle_part	Skupiny článku pro seskupeno dle stejného tématu
grouparticlecontentarchive	Klíčová slova v jednotlivých skupinách
grouparticlecontentnew	Klíčová slova v jednotlivých skupinách
itemrss	Tabulka pro RSS zprávy
keyworddisabled	Tabulka zakázaných slov
keywordsall	Tabulka pro klíčová slova
mainsource	Tabulka pro hlavní zdroje
pagerss	Tabulka RSS zdrojů
recoggroupsource	Tabulka pro rozpoznané skupiny katalogových zdrojů
resultdirectoryurl	Tabulka pro všechny odkazy katalogového zdroje
resultmainurl	Tabulka pro všechny odkazy hlavního zdroje
resultmainurldom	Tabulka DOM informaci každého odkazu
scanningpagerss	Tabulka skenování za poslední den
scanningpagerssarchive	Tabulka všech skenování RSS zdrojů

Databázové objekty 2

Procedura	Popis
ArchiveUnusable	Procedura pro přesun nepotřebných dat
countOrder	Funkce pro výpočet řazení skupin
createKeywodsTableFromArticle	Procedura, která prochází RSS zprávy bez obsahu
createKeywodsTableFromArticleById	Procedura, která získá klíčová slova z článku
recogGroupArticle	Procedura, která prochází RSS zprávy bez skupiny
recogGroupArticleById2	Procedura, která nalezne ke článku vhodnou skupinu
getNumDays	Funkce pro získání počtu dní
getNumHour	Funkce pro získání počtu hodin a minut
newScanPeriod	Procedura, která aktualizuje periodu skenování

Databázové objekty 3

Událost	Interval	Popis
eventbyday	1 day	Událost, která přesune nepotřebná data a provede aktualizace po skenování
createKeywodsTableFromArticle	20 minute	Událost, která spustí získávání obsahu
recogGroupArticle	20 minute	Událost, která spustí rozpoznávání skupiny

Pohled na objekty databáze:

ER diagram struktury databáze 1