

Кыргызский Национальный Университет им. Ж. Баласагына

Институт Интеграции Международных Образовательных Программ



КЫРГЫЗСКО-АМЕРИКАНСКИЙ ФАКУЛЬТЕТ
КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ИНТЕРНЕТ

УДК 681.3. (575.3) (043)

Похилько Андрей Федорович

РАЗРАБОТКА ПОИСКОВОЙ СИСТЕМЫ ДЛЯ ONLINE.KG

Выпускная работа на соискание академической степени бакалавра

Направление: 552801 Информатика и вычислительная техника

Специализация: Компьютерные информационные системы и Интернет

Выпускная работа рекомендована к защите

Декан КАФ-Интернет:
член-корр. НАН КР,
д.т.н., проф.

_____ Бримкулов У.Н.

« ____ » июня 2004 г.

Руководитель:
к.т.н., доцент

_____ Ямпольская С.А.

Нормоконтролер:
ст. преподаватель

_____ Большакова Т.Н.

Бишкек – 2004

Похилько А.Ф. КИС-4-00

РЕФЕРАТ

Выпускная работа: 58 с., 25 рис., 3 табл., 11 источников, 3 прил.

Ключевые слова: ПОИСКОВАЯ СИСТЕМА, ПОЛНОТЕКСТОВЫЙ ПОИСК, РЕЛЕВАНТНОСТЬ, ГИПЕРТЕКСТОВЫЙ ДОКУМЕНТ, ИНТЕРНЕТ, АНАЛИЗ НАВИГАЦИИ, ТРЕХМЕРНАЯ МОДЕЛЬ.

Цель работы – создание поисковой системы для Интернет.

В работе рассмотрены теоретические и практические аспекты разработки полнотекстовых поисковых машин и систем поиска информации в целом.

Разработанная система позволяет осуществлять индексацию и полнотекстовый поиск по указанной зоне Интернет с поддержкой гипертекстовых документов HTML, документов MS Word и простых текстовых файлов. Система автономна и осуществляет периодическую переиндексацию указанной зоны в зависимости от частоты обновления ресурсов. Большим плюсом системы является независимость от платформы и минимум требований к серверному программному обеспечению.

Рассмотрен процесс разработки системы на основе технологии PHP+MySQL и дополнительная технология VRML.

Дополнительно освещены моменты внедрения системы на разных серверах и приведены результаты их работы.

РЕФЕРАТ

Бүтүрүү иши: 58 бет, 25 сүрөт, 3 таблица, 11 адабият, 3 тиркеме.

Негизги сөздөр: ИЗДӨӨ СИСТЕМАСЫ, ТОЛУК ТЕКСТҮҮ ИЗДӨӨ, РЕЛЕВАНТТУУЛУК, ГИПЕРТЕКСТТҮҮ ДОКУМЕНТ, ИНТЕРНЕТ, НАВИГАЦИЯ АНАЛИЗИ, ҮЧ ӨЛЧӨМДҮҮ МОДЕЛЬ.

Иштин максаты - Интернет үчүн(!) издоо(!) системасын түзүү.

Бул иште толук текстүү издөө машиналарын жана жалпысынан маалымат издөө системаларын иштеп чыгуунун теориялык жана практикалык аспектилери каралган.

Иштеп чыгылган система HTML гипертекстүү документтерин, MS Word жана жөнөкөй текст файлдарын колдоо менен Интернеттин белгиленген аймагы боюнча индексация жана толук текстүү издөөнү жүзөгө ашырууга мүмкүндүк берет. Система автономдуу жана ресурстардын жазырттылуу жыштыгына жараша белгиленген аймакты мезгил-мезгили менен кайра индексация жүргүзүп турат. Системанын чоң артыкчылыгы платформадан көз каранды эместиги жана сервердик программалык камсыздоого талаптардын эң аз санда коюлушу болуп саналат.

PHP+MySQL технологиясынын негизинде системаны жана VRML кошумча технологиясын иштеп чыгуу каралган.

Системаны ар түрдүү серверлерге жайылтуу учурлары кошумча чагылдырылган жана алардын иш жыйынтыктары келтирилген.

ESSAY

Graduation work: 58 pages, 25 images, 3 tables, 11 sources, 3 attachments.

Keywords: SEARCH SYSTEM, FULL-TEXT SEARCH, RELEVANCY, HYPERTEXT DOCUMENT, INTERNET, NAVIGATION ANALYSIS, 3-D MODEL.

The purpose of work - creation of search system for the Internet.

Theoretical and practical aspects of full-text information search engine and search system development were considered in the work as a whole.

The developed system allows exercising indexation and full-text search on indicated internet zone, supporting HTML hypertext documents, MS Word documents and ordinary text files. The system is autonomous and executes periodical re-indexation of indicated zone depending on the frequency of renewal. The primary advantage of the system is independence from the platform and minimal requirements to server software.

The development of the system, based on PHP+MYSQL technology and supplementary VRML technology was considered.

The moments of application of the system on various servers were covered additionally and results of their operation were included.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	6
1. ТЕОРИЯ ПОИСКА В ИНТЕРНЕТ	8
1.1. Типы ИПС	8
1.1.1. Каталоги.....	8
1.1.2. Полнотекстовые	8
1.1.3. Мета-поисковые.....	9
1.2. Современные тенденции развития ИПС	9
1.3. Понятие релевантности	10
1.4. Этапы полнотекстового поиска и структура ИПС	10
1.4.1. Процедура индексации.....	10
1.4.2. Процедура поиска	11
1.4.3. Структура полнотекстовой ИПС.....	13
1.5. Алгоритмы поиска	15
1.5.1. Прямой поиск – мал, да удал.	15
1.5.2. Инвертированный файл	16
1.5.3. Алгоритмы суффиксных деревьев и сигнатур.....	16
1.6. Морфологический поиск – за и против	17
1.7. Требования к ИПС	18
1.8. Оценка результатов работы системы.....	18
1.9. Поисковая оптимизация ресурсов.....	19
1.9.1. Цель поисковой оптимизации и ее этапы.....	20
1.10. Проблемы, связанные с работой ИПС	20
1.10.1. "Черные дыры" в Интернет.....	20
1.10.2. Жулики.....	21
1.10.3. Загрузка каналов	21
1.10.4 Дубликаты документов	21
2. ПРОЕКТИРОВАНИЕ СИСТЕМЫ	23
2.1. Определение требований	23
2.2. Архитектура и принципы работы системы	23
2.2.1. Структура БД	24
2.3. Основные модули	24
2.3.1. Интерфейс к функциям БД	24
2.3.2. Модуль работы с URL	25
2.3.3. Пользовательский интерфейс администратора	25
2.3.4. Индексатор	26

2.3.5. Модуль поиска	30
2.4. Дополнительные модули.....	31
2.4.1. Автоматический тематический рубрикатор.....	31
2.4.2. Анализ гиперссылочной навигации.....	31
2.4.3. Трехмерная модель сайта.....	32
3. РЕАЛИЗАЦИЯ СИСТЕМЫ	33
3.1. Выбор технологий	33
3.1.1. PHP	33
3.1.2. MySQL	33
3.1.3. VRML.....	33
3.2. Программная реализация	33
3.2.1. Спиральный жизненный цикл и иерархия версий.....	33
3.2.2. Построение структуры БД.....	36
3.2.3. Написание программного кода.....	36
4. ВНЕДРЕНИЕ.....	38
4.1. Поисковый сервер для Kygnet.....	38
4.2. Поисковый сервер Kyrgyzstan On-Line.....	39
4.3. Студия NEW.....	42
4.4. Поиск для пользовательских сайтов.....	42
4.5. Логотип и название семейства ИПС.....	42
5. БУДУЩЕЕ СИСТЕМЫ	44
5.1. Недостатки разработанной ИПС.....	44
5.2. Шаг к распределенной обработке информации.....	44
5.3. Оптимизация запросов и структуры СУБД.....	44
5.4. Переход на другую СУБД.....	45
5.5. Улучшения морфологических функций.....	45
5.6. Развитие дополнительных модулей.....	45
ЗАКЛЮЧЕНИЕ	46
ГЛОССАРИЙ.....	47
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	51
ПРИЛОЖЕНИЯ	52
Приложение 1	52
Приложение 2.....	53
Приложение 3.....	54

ВВЕДЕНИЕ

Информатизация общества привела к повсеместному использованию электронных источников информации, которые, при большом накоплении знаний требуют средства быстрого поиска. Причем часто необходимо выполнить поиск не просто по названиям документов, а по полному тексту их содержимого. В этом случае на помощь приходят поисковые машины, в первую очередь, полнотекстовые. Разработка этих систем является одной из передовых областей современной информатики и требует слияния знаний в области лингвистики, информатики, программирования и проектирования хранилищ больших объемов данных и быстрой их обработки.

Целью выполнения данной работы является анализ принципов работы поисковых систем, как уже разработанных, так и исключительно теоретически описанных, проектирование и разработка на основе полученных данных поисковой системы для сервера Kyrgyzstan On-Line, удовлетворяющей требованиям современного поиска информации в Интернет.

Информационно-поисковая система (ИПС или просто поисковая система) – это ИС, позволяющая пользователю найти интересующую его информацию централизованно, без необходимости ручного перебора всех документов. ИПС известны нам с давних пор по примерам из библиотек, в которых были тематические и алфавитные каталоги, позволяющие найти интересующую книгу без утомительного перебора всех книжных запасов библиотеки. Также к ИПС можно отнести всяческие кадастры, реестры, регистры, применяющиеся в государственных учреждениях и на предприятиях для централизованного поиска информации в какой-либо предметной области. Конечно, эти системы не обладали вычислительными возможностями КИС, и библиотечную картотеку приходилось перебирать вручную, но современные компьютеризованные ИПС унаследовали все базовые принципы этих предков поисковых машин.

Первопричиной возникновения любой поисковой системы, компьютеризованной или нет, является переизбыток информации, среди которой нужно быстро находить интересующие человека документы. Далее в этой работе будут рассматриваться только компьютеризованные ИПС, в особенности системы полнотекстового поиска в Интернет.

Первой поисковой системой в Интернет был известный сервер Yahoo!, являвшийся как раз аналогом библиотечного тематического каталога для Интернет, и сразу завоевавший большую популярность среди посетителей всемирной паутины. Хорошим аргументом в пользу поисковых систем в Интернет является то, что два студента Стэнфорда (Дэвид Фило и Джерри Янг), создавшие эту первую поисковую систему в 1994 году, в 1995 уже стали миллионерами [1].

Цель ИПС – выдать пользователю набор результатов, наилучшим образом соответствующий его поисковому запросу, за минимальное время, и представление этого набора в наиболее информативном виде.

Далее мы увидим, что время поиска и адекватность результатов являются самыми значительными параметрами поисковой системы, а информативность представления и дополнительная функциональность системы является способом обогнать конкурентов на рынке поиска в Интернет, который на данный момент представляет серьезная армия ИПС.

Задачи информационно-поисковой системы:

- 1) сбор, обработка и хранение информации о документах;
- 2) прием запроса на поиск информации от пользователя;
- 3) поиск информации среди накопленных данных;
- 4) сортировка и выдача результатов поиска пользователю.

1. ТЕОРИЯ ПОИСКА В ИНТЕРНЕТ

1.1. Типы ИПС

1.1.1. Каталоги

Именно с этого типа ИПС началась история поисковых систем. Каталог – это система, в которой информация хранится структурировано по какому-либо признаку, чаще всего по предметной области. Как правило, информацию о новом документе, но, чаще всего, о целом наборе документов (сайте) в базу поисковой системы добавляет пользователь или администратор каталога, называемые иногда *экспертами*. Простейшим каталогом является карта сайта, которая дает возможность быстро найти документ в структурированном наборе ссылок.

Современные поисковые системы чаще всего имеют каталог или рубрикатор в дополнение к возможностям полнотекстового или мета-поиска, что позволяет сузить и конкретизировать область поиска.

Примеры каталогов в Интернет:

- 1) <http://www.yahoo.com/>;
- 2) <http://www.rambler.ru/>;
- 3) <http://www.ru/>;
- 4) <http://www.kg/>;
- 5) <http://www.online.kg/>.

1.1.2. Полнотекстовые

Поисковые системы этого типа появились с необходимостью поиска документов в стремительно расширяющемся пространстве Интернет, когда эксперты уже не успевают вносить в каталоги информацию о новых документах и удалять информацию о переставших работать ресурсах. Эти проблемы решают полнотекстовые поисковые системы, в которых занесение и проверка информации автоматизированы, и идут непрерывно на высокой скорости. Однако это является не только преимуществом, но и самым большим минусом полнотекстовых систем – только человек с его интеллектом способен принять полноценное решение относительно содержимого документа.

Примечание: Данная работа полностью посвящена именно полнотекстовым ИПС и их конструированию.

Примеры полнотекстовых поисковых систем:

- 1) <http://www.google.com/>;
- 2) <http://www.yandex.ru/>;
- 3) <http://www.aport.ru/>;

4) <http://www.rambler.ru/>.

1.1.3. Мета-поисковые

Мета-поисковые системы появились совсем недавно, когда количество и разнообразие ИПС привело к идее того, что можно послать запрос одновременно ко многим системам, сгруппировать результаты и выдать их пользователю. Это своеобразные надстройки над полноценными ИПС, обычно имеющие в своем списке полтора-два десятка лучших поисковых машин и выводящие в поисковом отклике информацию о найденном документе и поисковой машине, с помощью которой он был найден. Мета-поисковые системы позволяют соединять результаты от каталогов и полнотекстовых машин в едином списке ссылок [2].

1.2. Современные тенденции развития ИПС

Со времени первых поисковых систем прошло по меркам мира информационных технологий достаточно времени и ИПС изменялись постоянно, динамично подстраиваясь под веяния современных потребностей рынка информации. Современные тенденции развития поисковых систем характеризуются следующими признаками:

- 1) распределенность – для удовлетворительного функционирования ИПС нужны мощности суперкомпьютера, поэтому информация обрабатывается параллельно на сотнях и тысячах компьютеров;
- 2) смысловой поиск – в новейших концепциях ИПС поиск ведется не по буквальному набору слов, а по семантическому значению поискового запроса, что существенно расширяет диапазон документов, попадающих в результирующий набор;
- 3) тематический поиск – из-за колоссального объема информации пользователи имеют возможность вести поиск внутри какой-либо предметной области, которую иногда можно специализировать очень узко. Это позволяет избавиться от информационного мусора при поиске конкретной информации;
- 4) региональный поиск – появилась масса проектов регионального поиска, позволяющих сузить область поиска по географическому признаку;
- 5) поиск по разным типам документов – появление множества способов хранения информации и типов документов привело к тому, что современные поисковые системы позволяют находить не только гипертекстовые документы, но и файлы в формате PDF, DOC, XLS, PPT, искать изображения по ключевым словам и многое другое.

1.3. Понятие релевантности

Релевантность – это степень соответствия найденного документа поисковому запросу. Расчетная релевантность обычно выводится полнотекстовой системой исходя из информации, полученной в ресурсе: наличии ключевых слов, заголовков, выделений текста и т.п. Реальная релевантность документа чаще всего отличается от расчетной и может быть точно оценена лишь группой экспертов (людей). В этом преимущество каталогов перед полнотекстовыми системами – в каталоге эксперт может присвоить рейтинг документу, определив, таким образом, его релевантность, насколько это позволяет человеческий интеллект. Качество оценки релевантности является одним из ключевых параметров ИПС, так как при результирующем наборе ссылок в количестве десятков тысяч при поиске какого-либо часто встречающегося термина главной задачей становится их сортировка по убыванию релевантности. Проще говоря, если вы ищете по фразе «Президент Кыргызстана», то в первых строчках должен быть сайт Президента, полностью посвященный главе государства, а не заметка в Интернет-газете с единичным упоминанием о нем.

На реальную релевантность документа в значительной степени оказывают влияние не только и не столько внутрیدокументные факторы, такие как позиция термов, а внедокументные - позиция на сайте, индекс цитирования, частота обновления и т. п. Чем больше значащих факторов учитывает при оценке релевантности ИПС, тем качественнее будет сортировка результатов поиска.

1.4. Этапы полнотекстового поиска и структура ИПС

1.4.1. Процедура индексации

Индексация – это процесс поиска новых документов в Интернет и извлечения из них информации, достаточной для последующего нахождения этого документа с сохранением этой информации в хранилище, называемом *индексом*, отсюда название процесса.

Индексацию выполняет специальная программа – робот (паук, crawler). Первый робот был создан для того, чтобы обнаружить и посчитать количество веб-серверов в Сети, а затем подобные программы нашли свое прочное место в технологии полнотекстового поиска в Интернет. Одна из основных функций робота – запрос документов из сети с последующим извлечением находящихся в них ссылок и передачей содержимого документа обработчику текста. При этом нужно грамотно превращать относительные ссылки в абсолютные, учитывать протокол URL, тип документа,

некоторые другие параметры, чтобы предотвратить замусоривание списка ссылок заведомо ненужными элементами.

Обработчик текста занимается очисткой документа от тегов HTML, пустых символов, стоп-слов и т.п. Поле очистки документа результат передается на хранение в поисковый индекс, при необходимости изменяется для сохранения в нем. На рис.1 представлена структура индексатора.

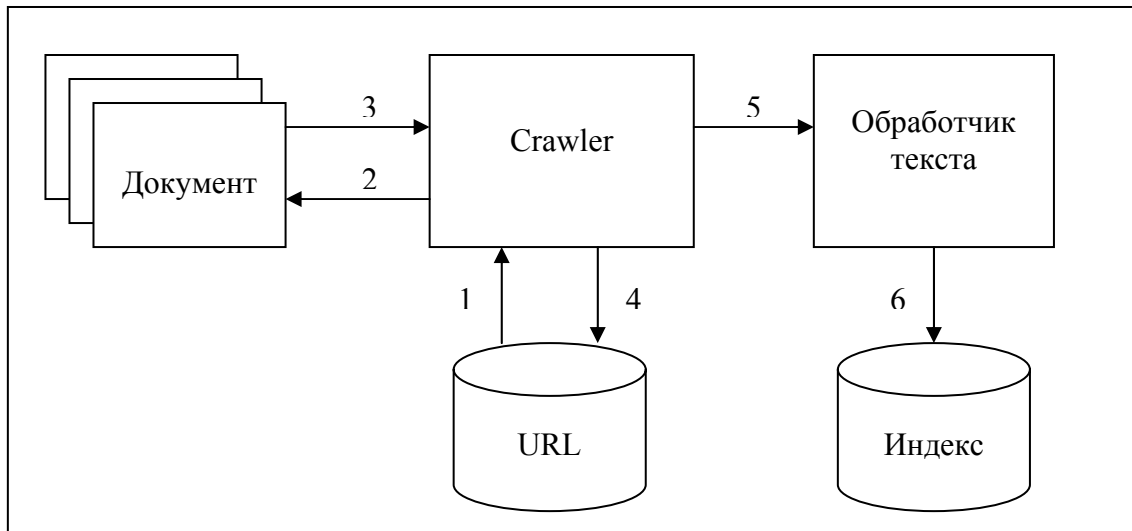


Рис. 1. Структура индексатора

- 1 – URL для индексации
- 2 – запрос на документ
- 3 – содержимое документа
- 4 – новые ссылки
- 5 – содержимое документа
- 6 – информация для сохранения в индексе

1.4.2. Процедура поиска

Процесс поиска - это всегда балансирование между двумя центрами тяжести: вывести как можно меньшее количество результатов, наиболее отвечающих запросу (релевантных), и, напротив, охватить наибольшее количество вариантов, включая словоформы, опечатки, и даже близкие по смыслу слова. Для наиболее широкого охвата документов используются всяческие морфологические, а иногда и семантические анализаторы, которые производят *леммизацию* и *стемминг* запроса для поиска по всем словоформам. Сама процедура поиска, как правило, уже оперирует не символьными данными, а идентификаторами морфологических групп, которые она получает от морфологического модуля. На выходе процедуры также идентификаторы документов, а не ссылки непосредственно. Объясняется такой подход тем, что числовая информация обрабатывается в компьютерных системах намного быстрее символьной.

Леммизацией называется процесс приведения слова к его начальной форме, а стемминг – это генерация всех словоформ из начальной. Вместе две эти процедуры позволяют в идеале охватить все морфологическое разнообразие языка.

Также крупные поисковые системы предоставляет пользователю язык логических операторов, позволяющий более конкретно описать искомый документ и текст в нем. В такой язык, как правило, входят операторы AND, OR, NOT, или их символьные аналоги, а также возможности группировки скобками, отметки цельных фраз кавычками и т.п.

Схематично процедура поиска представлена на рис. 2.

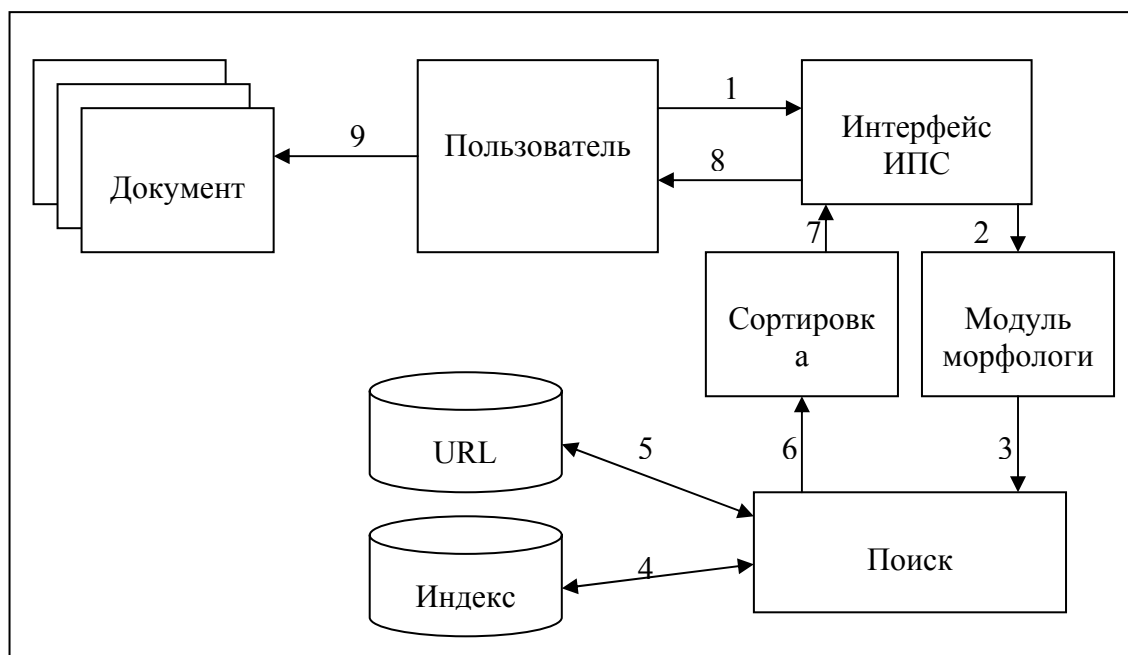


Рис. 2. Процедура поиска

- 1 – запрос поисковому интерфейсу ИПС
- 2 – слова запроса
- 3 – результат морфологического анализа
- 4 – поиск в индексе
- 5 – выбор ссылок на найденные документы
- 6 – информация о найденном
- 7 – сортировка по релевантности
- 8 – интерпретированные результаты
- 9 – запрос на документ

Чаще всего поисковую систему снабжают дополнительными элементами, предоставляющими расширения функциональности и улучшение качества работы. Например, вводится кэширование результатов поиска, которое позволяет не осуществлять полную процедуру поиска по одинаковым запросам, а брать готовые результаты из кэша. Также дается возможность найти похожие страницы, подсветить в найденном документе слова и прочие необязательные, но делающую систему весьма дружелюбной для пользователя функции.

1.4.3. Структура полнотекстовой ИПС

Структура ИПС (см. рис. 3) сильно зависит от того, на каком алгоритме основана ее работа, но, целом, в функциональной структуре полнотекстовой системы обязательно должны присутствовать следующие элементы:

- 1) индекс или иное хранилище данных, по которым ведется поиск;
- 2) интерфейс пользователя;
- 3) непосредственно поисковая машина.

Как мы увидим далее, индексатор может отсутствовать в составе системы (при алгоритме прямого поиска). Также может отсутствовать морфологический модуль и процедура сортировки. Однако эти допущения отрицательно сказываются на качестве работы системы (см.п. 1.10.).

1.5. Алгоритмы поиска

Алгоритмов полнотекстового поиска существует всего четыре, а широко используются лишь два них [3].

1.5.1. Прямой поиск – мал, да удал.

Алгоритм прямого поиска – это самое примитивное, что может прийти в голову разработчику. Суть его в том, чтобы просто перебрать имеющиеся данные и искать в них соответствие запросу без каких-либо предварительных извлечений информации. Этот алгоритм, несмотря на свою давнюю историю и простоту не теряет своей популярности среди разработчиков поисковых систем. Везде, где возможен перебор данных за разумный промежуток времени, можно применить прямой поиск. Если на сайте вашей фирмы всего пара десятков простых текстовых документов, то простой перебор их содержимого с поиском ключевой фразы будет наилучшим решением, и не надо тратить время и деньги на разработку сложной системы с предварительной индексацией.

Другое дело – поиск в масштабах Интернет, или, хотя бы, корпоративной сети. Разрозненные, разнотипные источники информации не позволяют быстро перебрать их и выдать результаты пользователю. К тому же представьте, что посетитель поискового сервера ввел запрос и поисковая машина лихорадочно устремила в Интернет в поисках результатов, перебирая *миллиарды* документов. Этот процесс будет идти не день и не два, а, скорее всего, никогда не угонится за постоянно изменяющейся всемирной сетью.

Простой пример – старая версия поисковой системы Google имела скорость индексации порядка 100 документов/сек. Даже в пространстве 1000 документов поиск прямым перебором занял бы 10 секунд. А ведь для качественного поиска нужны результаты поиска в пространстве, на несколько порядков большем.

Еще один минус прямого поиска – вычисление релевантности. Конечно, если нас удовлетворяет простой подсчет вхождений слова в документ как показатель релевантности, то все в порядке. Но, как сказано выше (см. п. 1.5.), на релевантность документа влияет большое количество факторов, которые при прямом поиске либо необходимо «на лету» вычислить, либо просто невозможно получить при таком алгоритме (например, количество документов, ссылающихся на найденный).

Но у прямого поиска есть серьезное преимущество перед поиском с предварительной индексацией – он гарантирует 100% актуальности информации. Как только изменение было внесено в страницу – оно автоматически попадает в поле зрения поисковой машины.

Второе существенное преимущество – создать подобную систему можно в короткие сроки и быстро адаптировать под конкретную задачу, другими словами,

инертность системы к изменениям минимальна. В индексирующую же систему вносить изменения рискованно, так как принципиальная смена какого-либо алгоритма требует *полной* переиндексации охваченного пространства, а это затраты времени, и, соответственно, денег.

Третье – легкая реализация языка запросов, который можно транслировать напрямую в логические конструкции поиска, так как информация хранится в исходном виде и не происходит ее потерь при индексации [3].

1.5.2. Инвертированный файл

Инвертированный файл – это структура данных, в которой хранится информация о том, в каких документах и, иногда, на каких позициях встречаются термины. Примером простейшего инвертированного файла может служить алфавитно-цифровой указатель, который обычно помещается в конце серьезных научных изданий. В этом указателе перечислены термины, и на каких страницах их можно найти. Подобное применяется в полнотекстовых поисковых системах, где в *индексе* сохраняется информация о том, какой терм, где встретился (иногда и в каком оформлении). Это порождает проблему хранения информации, так как подобная детализация в индексе требует вместительных хранилищ. Эту проблему решают применением записей не абсолютных позиций, а смещений от предыдущего, а также применением простых алгоритмов сжатия.

Применение сложных алгоритмов компрессии неэффективно, так как затраты вычислений на упаковку/распаковку записи могут замедлить работу системы и излишне нагрузить процессор.

Инвертированный индекс позволяет производить поиск с вычислением релевантности, основанной на многих факторах, как внутрیدокументных, так и внедокументных, так как алгоритм подразумевает предварительную обработку пространства поиска и позволяет собрать любую доступную информацию о ссылках, оформлении, структуризации ресурсов и т.п. Благодаря этому родились «умные» поисковые машины, в которых термины в документе ищутся по принципу их позиционной близости друг к другу, при сортировке результатов учитывается информация о ссылках на документ и из документа и масса прочих факторов.

В качестве минусов алгоритма можно назвать потери информации, которые возникают при помещении документа в индекс, ведь слова извлекаются из оригинального контекста.

1.5.3. Алгоритмы суффиксных деревьев и сигнатур

Неоднократно предлагались другие, отличные от инвертированного и прямого поиска алгоритмы и структуры данных. Это, прежде всего, суффиксные деревья, а также сигнатуры.

Первый из них функционировал и в Интернет, будучи запатентованным алгоритмом поисковой системы OpenText. Суффиксные индексы можно встретить и в российских поисковых системах.

Второй - метод сигнатур - представляет собой преобразование документа к поблочным таблицам *хэш-значений* его слов - "сигнатуре" и последовательному просмотру "сигнатур" во время поиска.

Широкого распространения ни тот, ни другой метод не получили [3].

1.6. Морфологический поиск – за и против

Морфологический поиск – это поиск с учетом языковых особенностей, например, поиск вне зависимости от формы, в которой стоит слово в документе и запросе, поиск по синонимам и другие возможности, которые дает лингвистический аппарат. Чаще всего морфологический поиск ограничивается приведением всех слов к их начальным формам и последующую генерацию всех возможных форм слова, а, затем, поиск по этим формам [4].

Часто морфологические функции реализуются в виде фиксированного словаря, в который включается как можно более широкий спектр терминов и слов из повседневного языка. Плюс такого подхода в хорошем контроле процедуры поиска и возможности однажды упорядочить словарь оптимальным образом и обеспечить быстрый поиск. Однако, фиксированный набор слов гарантирует, что в него не попадут новые слова, а в быстроменяющемся мире информационных технологий новые термины, названия продуктов и компаний появляются постоянно, и на них неизбежно появляется информационный спрос. Если ИПС не будет следить за обновлениями в языке, она очень скоро превратится в забытую.

Динамический словарь, в свою очередь, предоставляет массу головной боли разработчикам поисковых систем, так как вручную классифицировать не представляется возможным, а разработка универсального алгоритма невозможна, например, из-за омонимов в русском языке. Тем не менее, в последнее время были разработаны и внедрены подобные системы (например на ИПС Яндекс), и они успешно используют механизмы классификации неизвестных системе слов.

Вообще, применение морфологического поиска является одним из камней преткновения при разработке ИПС, ведь учет громадного морфологического разнообразия – это задача соответствующих вычислительных мощностей и времени. Даже поисковая машина №1 в мире Google долгое время отказывалась от морфологического поиска, аргументируя это тем, что хочет быть буквально отвечающей запросам пользователей, а не выводить результаты, в которых слова встречается в измененном виде.

К данному моменту морфологический поиск все же победил, и большая часть крупнейших систем конкурируют между собой в этой области, разрабатывая все более интеллектуальные алгоритмы морфологического анализа.

1.7. Требования к ИПС

Некоторые основные требования к современным крупным ИПС:

- 1) скорость индексации – 100-200 документов/сек как минимум;
- 2) размер – система должна иметь возможность обрабатывать любое количество документов;
- 3) период обновления - система должна обновлять информацию о состоянии и содержимом документа соответственно реальной частоте обновления ресурса;
- 4) указанные (submitted) страницы – система должна давать возможность добавить ссылку на документ, подлежащий индексации, вручную;
- 5) дата индексирования документа;
- 6) поддержка фреймов и ImageMap - система должна «понимать» и правильно обрабатывать применение этих вариантов оформления гиперссылочной навигации;
- 7) перенаправление (redirect) – система должна правильно обрабатывать перенаправления с одного URL на другой;
- 8) алгоритм определения релевантности – должен быть хороший алгоритм сортировки найденных ресурсов;
- 9) борьба с жуликами – система должна иметь эффективные средства борьбы с ложными ресурсами и попытками искусственных накруток позиций в результатах;
- 10) скорость поиска – система должна осуществлять поиск за время порядка 0,5 секунды даже по многословным запросам;
- 11) индексирование разнообразных типов документов (см. п. 1.4.);
- 12) удобный интерфейс – система должна представлять форму запроса и найденные результаты в понятном и удобном виде, давая возможность указывать дополнительные опции поиска и отображения.

1.8. Оценка результатов работы системы

Оценка результатов производится группой экспертов, которые оценивают в первую очередь следующие параметры:

- 1) точность (precision) – доля релевантного материала в ответе поисковой системы. Эксперты должны оценить, правильно ли система поняла смысл запроса и адекватен ли ее ответ. Оценивается качество сортировки результатов по релевантности. По современным данным, точность ИПС колеблется от 45% до 65%;
- 2) полнота (recall) – доля найденных релевантных документов в общем числе релевантных документов коллекции. Подобная оценка проводится, как правило, на специальных тестовых наборах данных. Запрашивается фраза, набор результирующих документов для которой в этой коллекции известен, и оценивается полнота поиска системы. Полнота зависит от качества распознавания ссылок в документах, потерь информации при занесении в индекс и морфологических возможностей поиска.

1.9. Поисковая оптимизация ресурсов

Развитие коммерческих ресурсов в сети, и ее расширение в целом привели к тому, что в откликах поисковых систем содержатся порой сотни тысяч документов. Исследования показывают, что посетитель поисковой системы обычно просматривает лишь первые 10-20 результатов поиска. Все это породило конкуренцию за верхние строчки в результатах поисковых систем и, как следствие, старания разработчиков документов добиться наиболее быстрой индексации ресурса с присвоением максимального коэффициента релевантности. Это было названо *поисковой оптимизацией*.

В общих чертах, для поисковой оптимизации разработчику ресурса необходимо не так уж много сведений, а именно:

- 1) как происходит наполнение базы данных ИПС и каков ее объем;
- 2) полный спектр возможностей поискового языка системы;
- 3) основные особенности представления результатов поиска, прежде всего, алгоритма ранжирования записей из списка отклика на поисковый запрос.

Проблема состоит в том, что современные ИПС, как правило, скрывают точные данные о деталях своей работы и остается лишь догадываться, как это происходит. Например, есть целое сообщество людей, которые пытаются догадаться, как же именно построена и работает система Google [5]. Во многих случаях можно получить приблизительное понятие о характеристиках системы несколькими запросами.

Как правило, в рамках поисковой оптимизации изменяются заголовки страниц, текст отмечается жирностью, выделяются заголовки в тексте и т.п.

1.9.1. Цель поисковой оптимизации и ее этапы

Цель поисковой оптимизации состоит в том, чтобы конкретная поисковая машина на конкретный запрос присваивала наибольший коэффициент релевантности ресурсу. Добиться верхних строчек в нескольких поисковых машинах весьма трудно из-за различий в оценке релевантности.

Поисковая оптимизация состоит из следующих этапов:

- 1) определение ключевых слов для ресурса;
- 2) анализ результатов в поисковых машинах по запросам, состоящих из ключевых слов и их комбинаций с выявлением сильных и слабых сторон;
- 3) изучение механизма подсчета релевантности конкретными поисковыми машинами;
- 4) разработка стратегии поисковой оптимизации ресурса с учетом результатов этапа 3;
- 5) ожидание переиндексации поисковыми машинами и переход к этапу 2.

Результатом поисковой оптимизации становится не только увеличение коэффициента релевантности для ресурса в поисковых машинах, но и улучшение логичности и информативности сайта, поскольку последние является первопричиной высокой релевантности ресурса.

1.10. Проблемы, связанные с работой ИПС

1.10.1. "Черные дыры" в Интернет

Первая, и одна из крупнейших проблем, с которыми сталкивается индексатор полнотекстовой ИПС при сканировании Интернет это зацикленная гиперссылочная навигация, также называемая «черными дырами» в Интернет. Причиной появления этого является плохая работа разработчиков Интернет-ресурсов.

При использовании маскирования файловой структуры сервера для обеспечения наиболее понятной навигации программисты используют перехват запросов на документ и обработку его вне зависимости от физического наличия запрашиваемого документа в файловой системе сервера. При этом чаще всего возвращается успешный результат в виде гипертекстового документа. Этот документ имеет ссылки, и если они поставлены неправильно, то могут указывать на уровень ниже вместо текущего, и, если обработчик все равно считает такие ссылки правильными, то происходит бесконечное зацикливание навигации. В результате количество документов искусственно возрастает почти до бесконечности, и, в большинстве случаев, этот рост происходит в геометрической

прогрессии. Естественно, такое явление способно «засосать» в себя поисковую машину, оккупировав ее фиктивными ссылками.

Примером зацикленной навигации может служить сайт <http://icn.online.kg/> , который порождает 8 фиктивных ссылок на каждый документ.

Также около 6 месяцев назад подобная проблема была обнаружена на сервере <http://www.aknet.kg/> , на данный момент ее наличие не проверялось.

Способы бороться с этой проблемой поисковые машины имеют несколько:

- 1) проверять URL документа, и отсекают документы с адресом, указывающим на зацикливание;
- 2) проверять содержимое документов – чаще всего они являются дубликатами;
- 3) ограничивать глубину индексирования навигации сайтов.

1.10.2. Жулики

Борьба за верхние строчки в откликах поисковых систем, как и в любой конкуренции, породила жуликов. Зачастую обладатели ресурсов пытаются нашпиговать документы громадным количеством ключевых слов, маскируя их под цвет фона, уменьшая шрифт и прибегая к другим ухищрениям. Если поисковая система не имеет средств распознавания жуликов, то она рискует потерять качество работы, выдавая в ответ на запросы пользователей горы «мусора», не содержащего полезной информации.

Например, поисковая система Яндекс долго грешила результатами, переполненными жульническими документами с рекламой, но, в последнее время, компания решила эту проблему.

1.10.3. Загрузка каналов

Еще одна проблема – загрузка каналов связи. Поисковые машины круглосуточно запрашивают документы с серверов, обеспечивая в некоторых случаях значительную загрузку каналов связи фиктивными запросами, среди которых зачастую становятся неразличимы посещения реальных пользователей.

К примеру, на сервере Kyrgyzstan On-Line доля запросов от поисковых машин доходит до 50%. Многие из них трудно идентифицировать, поэтому подсчет точной и реальной статистики посещений пользователей невозможен.

1.10.4 Дубликаты документов

Еще одна проблема, менее очевидная – это дубликаты содержимого. Доступность материалов в сети приводит к тому, что они зачастую копируются и дублируются в разных документах, изменяется лишь оформление страниц. Такую же проблему создают официальные и неофициальные зеркала ресурсов, которые хоть и не нарушают авторских прав, но, тем не менее, поставляют большое количество документов-дубликатов для

поисковых машин. Примером может служить проект Citfogum с десятками зеркал в разных регионах.

Та же система Яндекс довольно долгое время была масштабно критикуема за вывод большого количества дубликатов в результатах поиска. В настоящее время эта проблема компанией решена, но не полностью.

2. ПРОЕКТИРОВАНИЕ СИСТЕМЫ

2.1. Определение требований

Итак, определим требования к поисковой системе, которую нам необходимо разработать:

- 1) масштаб системы – около 20 000 документов (по оценкам файловой системы сервера www.online.kg);
- 2) обработка гипертекстовых документов и документов MS Word;
- 3) обработка фреймов и карт ссылок IMAGE MAP;
- 4) обработка всех вариантов адресации – абсолютной и относительной, с учетом протоколов и типов файлов;
- 5) возможность настройки зон индексации и параметров индексации для быстрого внесения изменений в характер работы ИПС;
- 6) скорость поиска в рамках допустимой (см. п. 1.9.);
- 7) сортировка результатов по релевантности;
- 8) возможность наблюдения за процессом индексации и подсчет статистики;
- 9) обеспечение морфологического поиска хотя бы на минимальном уровне;
- 10) представление результатов в простой и информативной форме с подсветкой найденных вариантов;
- 11) минимум требований к конфигурации сервера. Система должна требовать для работы только наличия PHP и MySQL. Требования установки сторонних программ и привязка к операционной системе недопустимы;
- 12) разработать эффективный механизм переиндексации пространства, так, чтобы часто обновляемые ресурсы индексировались чаще;
- 13) борьба со всеми вышеперечисленными проблемами работы ИПС.

2.2. Архитектура и принципы работы системы

В качестве рабочей среды и основной технологии выбирается *Веб*, и это логично ведет к принципу работы в *итерациях*. Такая работа имеет ряд преимуществ перед непрерывной работой. Во-первых, система имеет между итерациями некоторую «передышку», в которой можно безбоязненно остановить процесс индексации без риска нарушения целостности БД, а, также, если система используется на аппаратной платформе, применяющейся для еще каких-то задач, эти задачи имеют возможность выполняться наряду с индексатором. Во-вторых, при изменении параметров или настроек

системы, они вступают в действие на следующей итерации, и нет необходимости перезапускать процесс индексации.

Минусы итерационного подхода – затраты времени на инициализацию на каждой итерации, простой системы между итерациями.

Принято решение генерировать итерации с периодом 1 минуты, а длительность итерации не более 45 секунд. Это обеспечит достаточную загрузку системы, но перерывы между итерациями также достаточны для действий обслуживания. Также решено ввести блокировку индекса на время работы, чтобы предотвратить запуск параллельного индекса, если первый еще не завершил работу, так как это приведет к неоднозначности результатов.

2.2.1. Структура БД

На рис. 4 представлена структура БД для проектируемой системы.

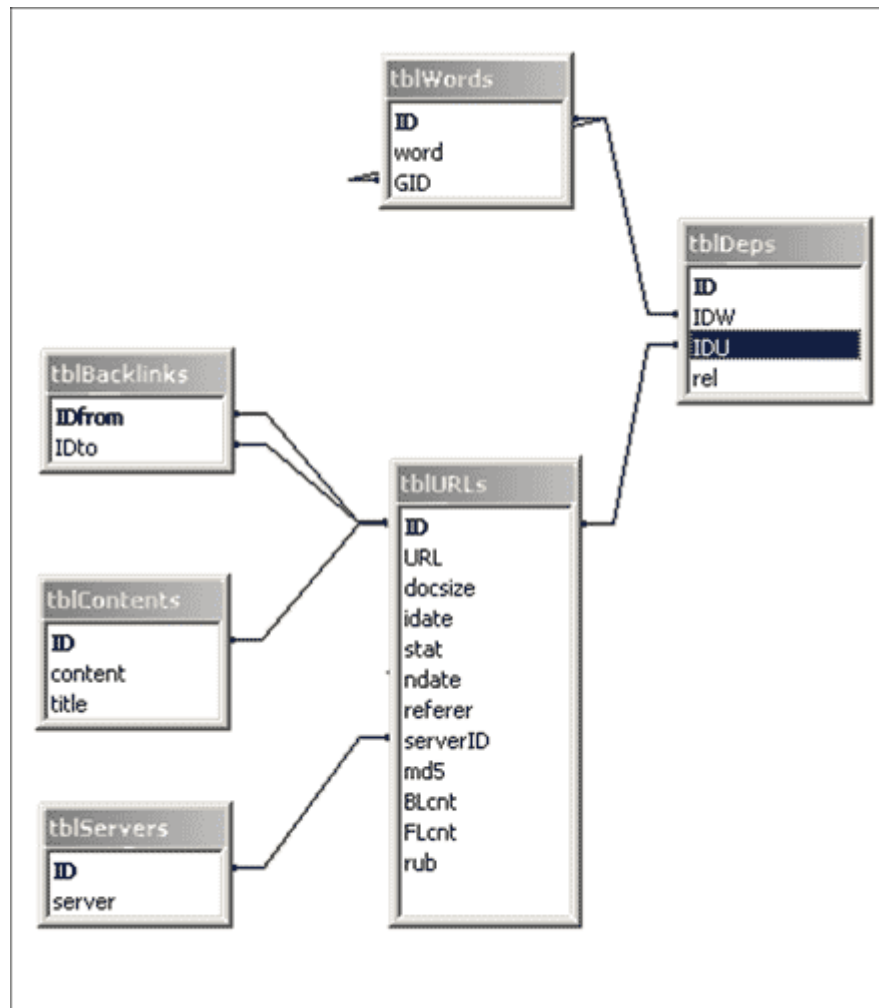


Рис. 4. Структура БД для проектируемой ИПС

2.3. Основные модули

2.3.1. Интерфейс к функциям БД

Работа с базой данных, в которой хранится вся информация для работы поисковой машины, занимает одно из центральных мест. Поэтому необходимо продумать работу с

БД самым тщательным образом. Для работы с БД необходимо разработать специальный модуль, через который будет производиться взаимодействие с хранилищем данных.

Функции модуля:

- 1) отслеживание ошибок при работе с БД, и остановка программы при их появлении, если не указано обратное;
- 2) ведение журнала запросов SQL к БД для отладки и оптимизации работы;
- 3) быстрый и наглядный вывод результатов выборок в виде таблиц.

2.3.2. Модуль работы с URL

Также для ИПС необходим полный набор функций работы с URL. От качества этой работы зависит то, насколько полно система проиндексирует зону поиска.

Функции модуля:

- 1) приведение относительных URL к абсолютным;
- 2) проверка на заикленность навигации по URL;
- 3) исключение дублирования адресов за счет наличия или отсутствия указателя службы WWW (www.kaf-i.kg и kaf-i.kg должны рассматриваться как один и тот же сервер);
- 4) проверка расширений запрашиваемого документа на соответствие индексируемому типу;
- 5) проверка вхождения URL в зоны поиска;
- 6) очистка URL от ненужных параметров вроде идентификаторов сессий.

2.3.3. Пользовательский интерфейс администратора

Пользовательский интерфейс администратора не должен обладать большим набором функций и нет нужды затрачивать много ресурсов на его удобство и красоту. Причина в том, что система рассчитана на автономность, и необходимы лишь базисные функции, такие как:

- 1) возможность запустить индекатор в ручном режиме и просмотреть его работу с соответствующими комментариями происходящего;
- 2) просмотр статистики и текущего состояния ИПС;
- 3) исполнение операций проверки целостности БД.

На рис. 5 представлен вид интерфейса администрирования. Ссылки управления находятся вверху и представлены в виде простых слов. Остальную часть занимает статистическая информация о ИПС.

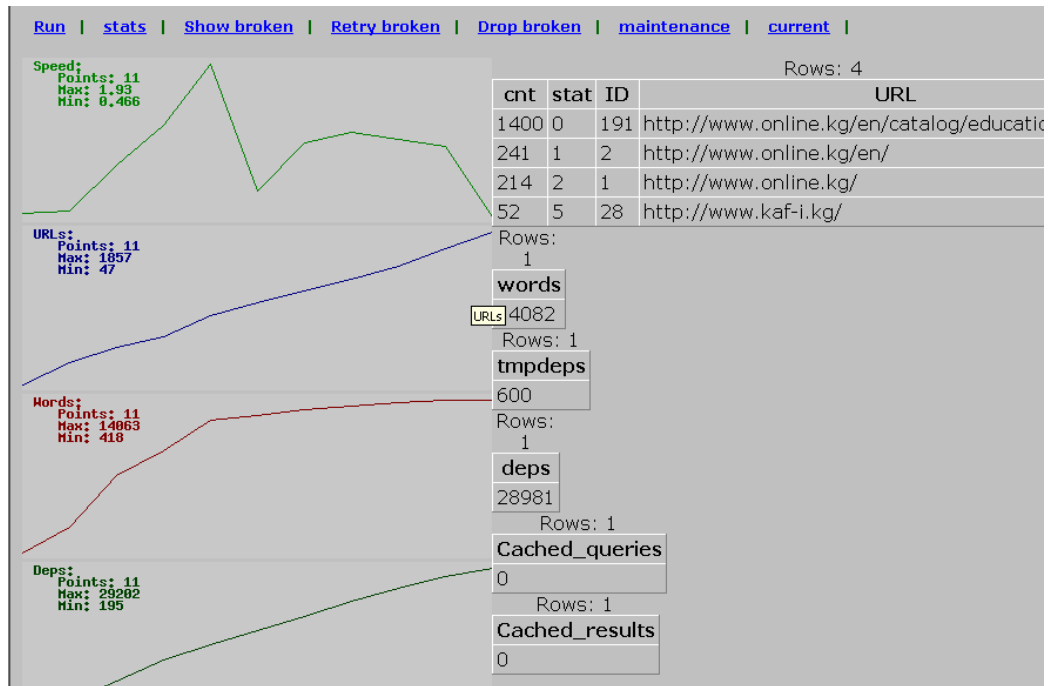


Рис. 5. Вариант интерфейса администратора ИПС

2.3.4. Индексатор

Собственно индексатор является сердцем системы и все основные характеристики ИПС зависят от качества и скорости его работы. Индексатор разделяется на несколько подсистем, которым передаются результаты работы его основной части. Функции основы индексатора:

- 1) получение содержимого документов по адресам URL;
- 2) сканирование документа и выделение ссылок;
- 3) обработка ссылок через модуль работы с URL и постановка их в очередь;
- 4) передача документа и его параметров в подсистемы;
- 5) сохранение результата индексации, например, даты последней индексации, ее кода завершения и размера файла в базе URL;
- 6) сохранение полного текста документа в специальной таблице для будущего отображения цитат в найденных документах;
- 7) отсека дубликатов документов по чистому содержимому.

На рис. 6 изображена информация, которую может получить администратор о процессе индексации.

```

1. http://www.online.kg/en/catalog/lifestyle/4/ : 200
new 1.431: make index 1
0.0002: md5
0.2093: новых ссылок: 10/117
s+ Назначено: 29.06.2004 [20:49]
0.0046: make index 2
0.2442: Words:178/64/51
0.0002: 64: 36%
0.0012: Размер: 21627 / 904

2. http://www.online.kg/ru/go/230/ : 302

3. http://www.online.kg/ru/go/228/ : 302

4. http://www.online.kg/ru/go/236/ : 302

5. http://www.online.kg/ru/go/38/ : 302

6. http://www.online.kg/ru/catalog/international/2/ : 200
new 1.6225: make index 1
0.0002: md5
0.189: новых ссылок: 6/113
s+ Назначено: 29.06.2004 [20:49]
0.0048: make index 2
0.3247: Words:280/116/86
0.0001: 116: 41%
0.0017: Размер: 21444 / 1599

```

Рис. 6. Информация о процессе индексации

2.3.4.1. Фильтр текста

Фильтр текста получает на входе полный документ от главной части индексатора. Задача фильтра – очистить текст от всяческого «мусора», который будет занимать лишнее место. Чем меньше лишней информации будет попадать в поисковый индекс, тем быстрее будет вестись по нему поиск.

Фильтр делает свою работу в два этапа. На первом этапе из документа удаляется все, что относится к гипертекстовой разметке и программированию. Также делается замена мнемоник и специальных символов на безопасные в смысле ASCII символы. Результат первого этапа сохраняется в таблицу содержимого документов. Сделано это для того, чтобы пользователь мог при поиске увидеть ситуацию, в которой встретился искомый терм, включая знаки препинания.

Второй этап фильтрации убирает все знаки препинания вообще. Результат передается подсистеме словарного и морфологического разбора.

2.3.4.2. Модуль словарного разбора и морфологии

От морфологического модуля зависит не только качество поиска, но и его скорость. Одна из основных функций морфологического анализа – группировка однокоренных слов, так называемая, *лемматизация*. Группировка позволяет не сохранять в БД информацию о встрече в документе всех форм слова, а только его основы.

Основная проблема – это группировка слов. Делать ее вручную не представляется возможным, так как, по оценкам экспертов, масштабы словарей порядка сотен тысяч слов, а если прибавить к этому опечатки, ошибки и прочие случайные совпадения, то прибавляется еще несколько десятков тысяч записей. По этому необходимы средства хотя бы минимальной автоматизации этого процесса. Однако возможности ручного редактирования тоже нужны, так как возможные проблемы непредсказуемы.

На рис. 7 представлен пример ручного интерфейса управления словарными группами. Налицо обилие функций и возможностей, в противоположность аскетичному меню администрирования индекса.

	m%
13751	m
15250	maaian
15059	mabila
12663	macburger
1996	machine
1997	machines
8924	machu
13932	mackarova
1689	made
1187	madinina
938	madonna
13094	madumarov
13048	magazine
1554	magazines
539	mail
1519	mail333

Рис. 7. Управление словарной базой ИПС

Для автоматизации был найден проект RiSearch, в котором эти функции частично реализованы и проект был взят за основу [6].

Итак, в первую очередь нужно осуществить словарный разбор поступившего отфильтрованного документа. Так как знаки препинания в нем уже отсутствуют, достаточно простого разбиения по пробелам.

Функции словарного разбора:

- 1) разбиение входящей строки на слова по пробелам;
- 2) исключение символов, не относящихся к буквам и цифрам;
- 3) определение, на каком языке написано слово и коррекция символов, пишущихся одинаково в разных языках.

Функции морфологии:

- 1) приведение всех слов во входящем наборе к их начальным формам;
- 2) группировка слов с подсчетом количества вхождений каждого слова в набор;
- 3) запись в индекс информации, о вхождении данного слова в этот документ и его вес для этого документа.

2.3.4.3. Поддержка файлов MS Word

Так как в сети Интернет множество полезных ресурсов представлено в виде документов MS Word, было бы крайне полезным поддерживать поиск по этому типу

файлов. Однако, файлы DOC имеют очень сложный формат бинарных потоков и для их перекодировки в аналог гипертекста требуется установка дополнительного программного обеспечения на сервер, что лишает систему переносимости.

По этим причинам автор произвел анализ формата файлов Word путем простого просмотра бинарного файла в текстовом редакторе. Было замечено, что во многих случаях текст в документе можно идентифицировать по признакам окружающих его кодов. Так, выяснилось, что при простом сохранении файла русские символы предваряются шестнадцатичным кодом 4, а английские имеют перед собой символ с кодом 47. Это позволило написать простенький алгоритм, вычлняющий из документов Word текст. Конечно, это происходит не всегда и не полностью, но результаты удовлетворительны.

2.3.4.4. Графики работы

Также было бы весьма полезно визуализировать динамику работы ИПС, для отслеживания общих тенденций наполнения баз данных. Для этого планируется сделать набор графиков, отображающих визуально процесс работы поисковой машины и изменения наполненности индекса и БД. Ожидаются следующие закономерности:

- 1) Наполнение списка URL должно происходить примерно по закону, представленному на рис. 8.

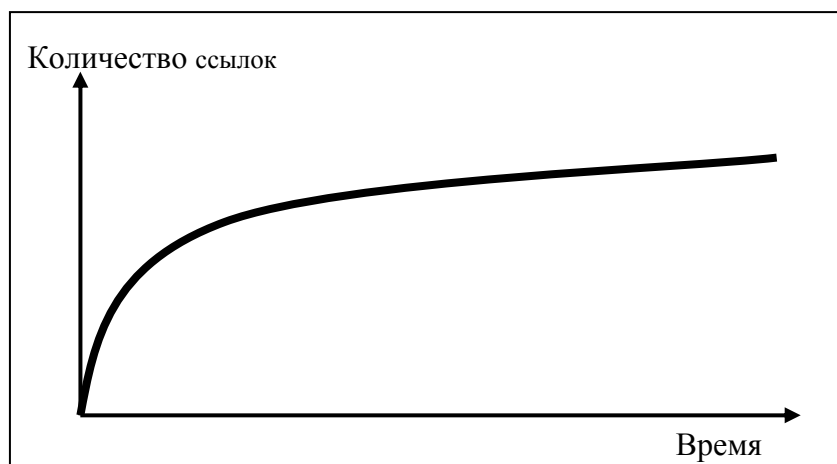


Рис. 8. Ожидаемая динамика таблицы ссылок и словаря в БД

То есть, количество URL возрастает интенсивно в начале индексации, но постепенно новых данных становится все меньше, список приходит в состояние насыщения.

- 2) Состояние таблицы словаря будет изменяться, как ожидается, примерно по такой же закономерности, что и URL.
- 3) Поведение таблицы индекса предсказать сложно, так ее динамика зависит от наполнения таблиц URL и словаря, но, поскольку обе этих таблицы примерно, повторяют динамику друг друга, то можно сделать вывод, что интенсивность наполнения индекса примерно линейна и, предположительно, зависит от

отношения количества встретившихся слов к количеству просмотренных документов (см. рис. 9).

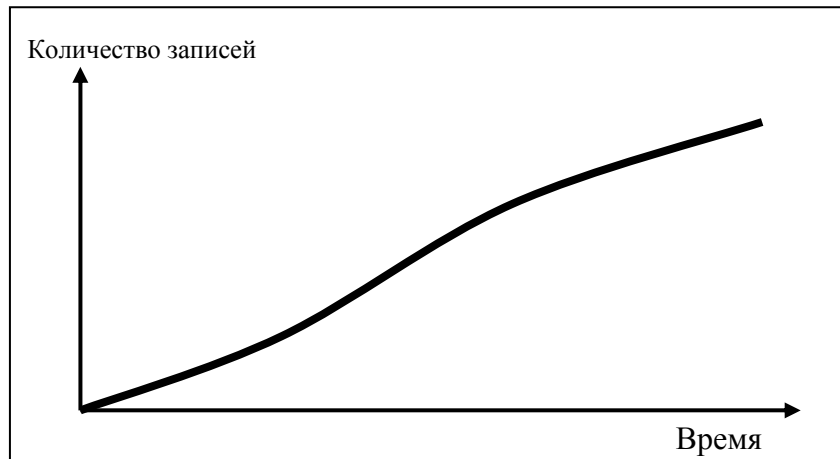


Рис. 9. Ожидаемая динамика таблицы индекса в БД

- 4) Скорость индексации (см. рис. 10) по замыслу должна естественным образом уменьшаться по закону обратной пропорциональности, то есть по гиперболе, так как затраты на добавление данных и перестройку индексов в процессе индексации будут возрастать и скорость индексации будет падать, пока не дойдет до предельного значения, при котором ИПС вообще не сможет производить операции с такими объемами данных.

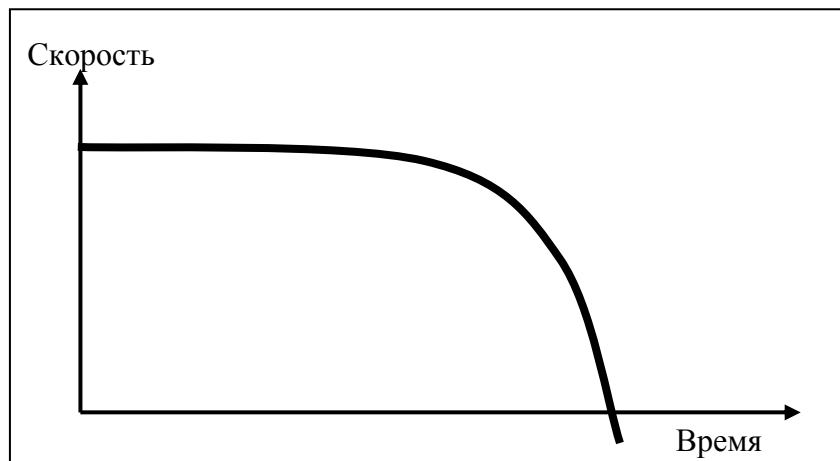


Рис. 10. Ожидаемая динамика скорости индексации ИПС

2.3.5. Модуль поиска

Вторым центром тяжести в ИПС является подсистема поиска по индексу. Эта подсистема так же, как и индексатор, исполняет функции, оказывающие критическое влияние на оценки работы поисковой системы. От модуля поиска требуется, прежде всего, точность, затем скорость работы.

Функции модуля поиска:

- 1) предоставление пользователям интерфейса, позволяющего сделать запрос;
- 2) прием запроса и фильтрация текста запроса;
- 3) разбиение запроса на отдельные слова;
- 4) поиск набора слов в индексе;
- 5) сортировка результатов;
- 6) интерпретация и возвращение результата пользователю;
- 7) предоставление удобного просмотра списка результатов с разбиением его на страницы.

2.3.5.1. Кэширование результатов поиска

Для снижения излишней загруженности сервера было бы неплохо ввести кэширование результатов поиска, чтобы при частом поиске по одинаковому запросу, например, при простом перелистывании страниц результата, не происходил фактический поиск по индексу, а брался готовый результат из кэша. Это значительно ускорит работу системы и снизит ее загруженность.

2.4. Дополнительные модули

Для расширения базовой функциональности планируется разработка и подключение дополнительных модулей для ИПС.

2.4.1. Автоматический тематический рубрикатор

Тематический рубрикатор является дополнением к полнотекстовой ИПС, которое снабжает ее частью функциональности системы поиска типа *Каталог*. Существенное различие в том, что в предлагаемом рубрикаторе наполнение должно происходить автоматически. Естественно, необходимо как-то определять, к какой рубрике должен относиться документ. В данный момент принято решение делать это на основании заголовка документа и наличия в нем ключевых слов для данной рубрики.

2.4.2. Анализ гиперссылочной навигации

Собранная индексатором информация о ссылках между документами позволяет произвести анализ графа документов и найти проблемные места в гиперссылочной навигации. Существует способ рассчитать все кратчайшие пути в графе, представленном в виде матрицы, и это позволит указать места, где этот граф имеет слишком длинные пути, а где он их вообще не имеет. Следует отметить, что граф направленный и циклический, что усложняет расчеты. Также разработана формула расчета оценки гиперссылочной навигации, которая поможет оценить качество разработки ссылок между документами.

Формула выглядит следующим образом:

$$k = \left(1 - \frac{P_{\max}}{n}\right) \cdot \frac{K_{ideal}}{K_{real}},$$

где k – это оценка навигации, $0 \leq k \leq 1$;

P_{\max} – максимальный путь в графе;

n – количество вершин в графе;

K_{ideal} – проходимость для идеального (полносвязного) графа такой же размерности;

K_{real} – проходимость для реального графа.

Оценка абсолютна. Расчет ведется из следующих соображений:

- 1) оценку 1 должен иметь полносвязный граф;
- 2) чем больше максимальный путь в графе, тем ниже его оценка;
- 3) чем больше узлов в графе, тем большие максимальные пути допустимы.

Безусловно, расчетная формула требует доработки, но даже в таком виде она учитывает достаточно факторов гиперссылочной навигации и позволяет сравнить любую группу документов, чаще всего, группируемых по сайту, на предмет качества навигации.

В качестве дополнительных возможностей можно отобразить списки самых длинных, а, следовательно, проблемных путей в графе, и список невозможных переходов от документа к документу по гиперссылкам.

2.4.3. Трехмерная модель сайта

Также появилась интересная идея построить на основе информации о документах и ссылках между ними трехмерную модель, которая должна дать возможность просмотра одновременно всей структуры сайта. Такую возможность дает только карта сайта, но она не отображает *фактическое* наличие ссылок между документами, а лишь логическую их систему. Разрабатываемый модуль должен визуализировать реальные документы и ссылки между ними.

3. РЕАЛИЗАЦИЯ СИСТЕМЫ

3.1. Выбор технологий

3.1.1. PHP

Поскольку основной средой разрабатываемой ИПС будет Веб, выбор безусловно пал на язык серверных сценариев PHP. Причины этого в следующем:

- 1) автор лучше всего знает этот язык;
- 2) отличная переносимость. PHP одинаково реализуется как на UNIX- так и на Windows-системах;
- 3) ориентация языка на обработку символьной и строковой информации;
- 4) скорость разработки;
- 5) современные методы программирования, включая ООП.

3.1.2. MySQL

В качестве СУБД выбран бесплатно распространяемый сервер баз данных MySQL. Он обеспечивает хорошее быстродействие и достаточные объемы данных. Также MySQL хорошо взаимодействует с PHP.

3.1.3. VRML

Для трехмерного моделирования (см. п. 2.4.3.) выбрана технология VRML. VRML – это стандартизованный язык разметки трехмерных миров или сцен. VRML – это технология в Интернет, поэтому построенные модели можно просматривать в режиме онлайн. Это повышает практическую ценность модуля трехмерного моделирования, как средства привлечения посетителей. [7] [8]

Большой плюс этой технологии в том, что данные передаются как и гипертекстовые документы – текстовым форматом, а не графическим. К тому же программы для просмотра таких сцен (VRML-браузеры) встраиваются в обычные программы просмотра Интернет и позволяют произвольно перемещаться в пространстве, вращая сцену как вздумается пользователю.

3.2. Программная реализация

3.2.1. Спиральный жизненный цикл и иерархия версий

От появления идеи разработки этой поисковой системы до нынешнего момента (30 июня 2004 г.) прошло около полутора лет. Естественно, что реализация системы не велась сразу к конечному продукту и было множество промежуточных версий программы.

В целом, работа происходила соответственно модели *спирального жизненного цикла*, который неотъемлемой является частью технологии RAD (см. рис.11). Это позволило не только быстро разработать систему, но и породить в процессе разработки ряд самостоятельных и законченных версий, которые даже были успешно внедрены. [9]

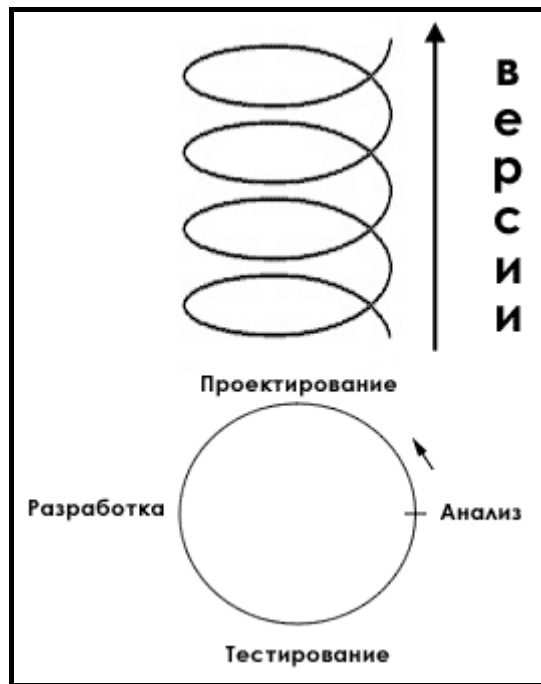


Рис. 11. Спиральный жизненный цикл ИПС

В целом, на данный момент идет 13 виток спирали. В процессе разработки происходило разветвление версий (см. рис. 12), когда на следующем витке жизненного цикла разрабатывалась не одна, а две системы. Иногда даже происходил откат к принципам предыдущих версий. С появлением версии, ныне названной NEW, было принято решение начать собственно отслеживание истории развития версий системы, так как стало понятно, что она будет развиваться долго и разнообразно.

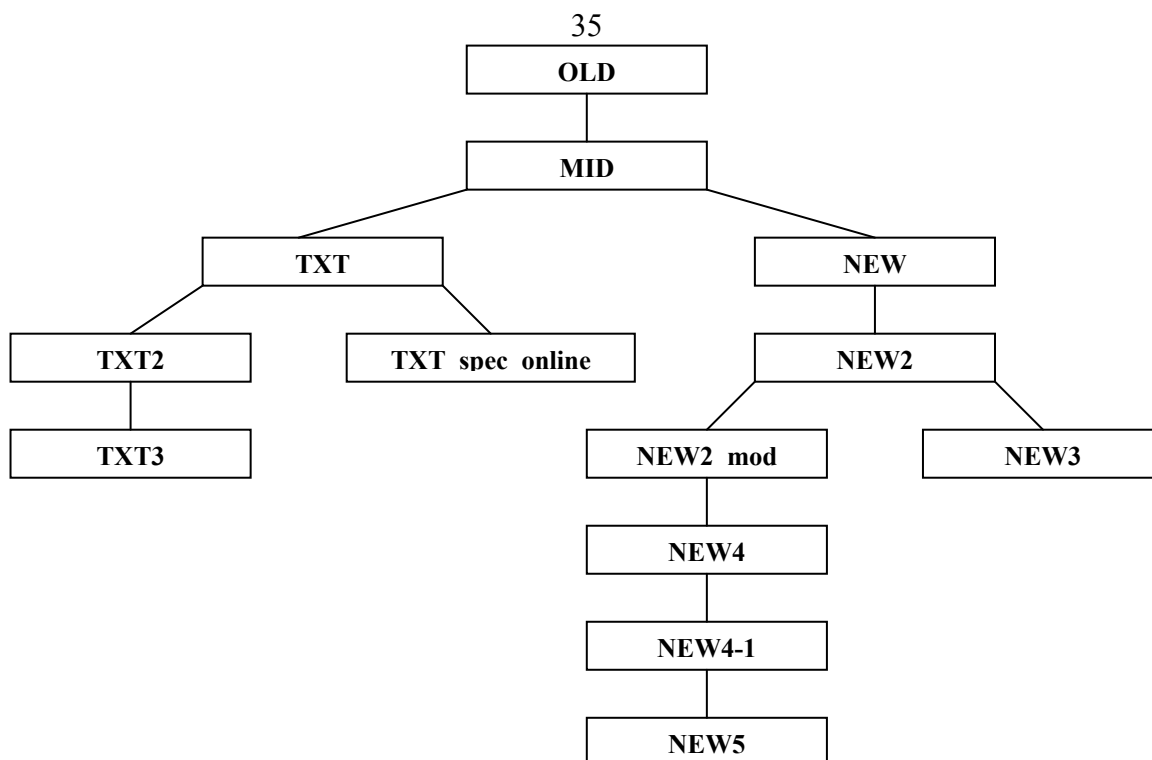


Рис. 12. Иерархия версий ИПС

Таблица 1

Комментарии по версиям ИПС

<i>Версия</i>	<i>Комментарий</i>
OLD	Начальный, простой БД – MySQL, одна таблица
MID	Новая версия простого, с использованием встроенного подсчета релевантности функциями MySQL и группировкой по серверам Пара-тройка таблиц БД MySQL
TXT	Простой. Не отлаженный; БД - Текстовый файл
TXT2	Легкие исправления для студии НБЮ
TXT3	Капитально отлажен для студии НБЮ
TXT_spec_online	Спецверсия, встроенная в оболочку online.kg
NEW	Переход на связи URL-слово Отсутствие языкового модуля словоформ
NEW2	Отладка системы слова-URL Тестирование быстродействия и неудовлетворительные результаты
NEW3	Попытка возврата к полнотекстовому поиску через функции MySQL. Провал
NEW2_mod	Модуль словоформ Выделение модулей, конфигурационных файлов
NEW4	Вынос содержимого документов в отдельную таблицу, кэш запросов, временные таблицы для ускорения
NEW4-1	Добавление дополнительных индексов на таблицах и ускорение работы. Оптимизация по времени всех программных кодов.
NEW5	Переделка паука на класс; Графики работы; Индексы цитирования Класс рубрикатора; Очередное ускорение и сборка в кучу

3.2.2. Построение структуры БД

На данный момент реальная структура БД отличается от спроектированной в начале и описанной в п. 2.2.1. Добавились дополнительные таблицы, осуществляющие кэширование входных и выходных данных ИПС для ускорения ее работы, таблица модуля автоматической рубрикации.

Также значительный вес для системы имеет расстановка индексов на полях таблиц БД, по которым СУБД ведет поиск внутри таблиц. Правильная расстановка индексов способна значительно ускорить выборку из таблиц, но, вместе с тем, индексы замедляют изменение и добавление данных в таблицу, так как СУБД приходится их перестраивать. В этом случае помогают промежуточные таблицы, в которые попадают данные, и которые через определенные промежутки сбрасывают накопленное в главные таблицы.

3.2.3. Написание программного кода

Написание программного кода, как сказано выше, заняло около полутора лет. При индивидуальном программировании трудно бывает систематизировать работу, порой модификации кода и фиксация возникших идей происходят что называется «на коленке». Разработка поисковой системы не планировалась как тема будущей дипломной работы, поэтому точных данных о том, как происходил этот процесс, автор не имеет. Тем не менее, отслеживается четкая закономерность: как только какая-либо подсистема разрастается и выделяется в системе, она выносится в отдельный файл.

Программный код разбит на файлы по назначению, поэтому все преимущества модульного программирования используются в полной мере.

Существует три файла, через которые осуществляется запуск системы: поисковый, индексатор, и дополнительный. При запуске любого из этих файлов происходит подключение соответствующего ему набора модулей и вызов функций этих модулей. Попытка запуска отдельно модуля ничего не даст, так как в нем хранится только описание функций, а вызовов нет.

Пример программного модуля (модуль кэширования результатов поиска):

```
<?
function cch_mq($q)
{
    global $tblCQ,$tblCR;
    $lim=time();
    if ($q[0]!==' ')
    {
        return db_mq("select IDU,rel,cnt from $tblCR where IDQ=".$q." order by cnt desc, rel
desc");
    }
    else
    {
```

```

        echo "&nbsp; &nbsp;";
        return false;
    }
}

function cch_get_idq($q)
{
    global $tblCQ,$tblCR,$query_expires;
    $lim=$query_expires;
    $res=db_mq("select ID from $tblCQ where query=\"$q\" and expires>".time().".");
    if (mysql_num_rows($res))
    {
        $tmp=mysql_fetch_row($res);
        $ID=$tmp[0];
    }
    else
    {
        $tmp=mysql_fetch_row(db_mq("select ID from $tblCQ where query=\"$q\""));
        db_mq("delete from $tblCQ where ID=".$tmp[0],true,true);
        db_mq("delete from $tblCR where IDQ=".$tmp[0],true,true);
        db_mq("insert into $tblCQ values(0,\"$q\","."(time()+$lim).")",true,true);
        $ID=" ".mysql_insert_id();
    }
    // db_mq("update $tblCQ set expires=".(time()+$lim)." where ID=$ID");
    }
    return $ID;
}
?>

```

4. ВНЕДРЕНИЕ

В виду практического отсутствия поисковых серверов в кыргызском Интернете и реальной потребности в поиске по выросшим объемам отечественных гипертекстовых документов, внедрение разработанной системы не сталкивалось с особыми трудностями.

4.1. Поисковый сервер для Kyrnet

Внедрение поисковой системы версии ТХТ происходило на соответствующем витке жизненного цикла, и было одним из первых попыток опробовать систему в реальных условиях. Хорошую проверку прошла система на предмет переносимости и перенастраиваемости. Был зарегистрирован домен arc.gratis.kg, куда была установлена ИПС версии ТХТ, и настроена на поиск по зоне to.kg и gratis.kg. Единственной проблемой оказалось инициирование итераций индексатора, но эта проблема была решена запросами с сервера online.kg. В последствии, Тургумбаев Бакыт предложил разработать дизайн для этой системы, который и был принят (см. рис. 14).

Не смотря на то, что по правилам хостинга www.kyrnet.kg этот сайт необходимо удалить, администраторы оставили его, возможно, из-за его уникальности для сервера.

На данный момент ИПС содержит в своей базе около 3 000 документов, по которым успешно осуществляет поиск (см. рис. 13).

```
1. 1150 http://kate.gratis.kg/main.php?id=1406 cr parsed 0 sec. Size: 2875 / 11387 new links: 1/41
2. 1151 http://kate.gratis.kg/main.php?id=1402 cr parsed 0 sec. Size: 426 / 7764
3. 1152 http://kate.gratis.kg/main.php?id=1403 cr parsed 0 sec. Size: 931 / 7020
4. 1153 http://kate.gratis.kg/main.php?id=1404 cr parsed 1 sec. Size: 1207 / 9507
5. 1154 http://kate.gratis.kg/main.php?id=1405 cr parsed 1 sec. Size: 582 / 7655
6. 1155 http://kate.gratis.kg/main.php?id=1701 cr parsed 1 sec. Size: 1959 / 7997
7. 1156 http://reflection.to.kg/html/help.html unable to retrieve
```

Рис. 13. Процесс индексации на arc.gratis.kg

search engine

точно сказать Search

Search rule: OR AND Exact phrase

- [Наск-Куль >> Статьи >> Как стать хэкером](#)
... в конечном итоге сводящимся к одному: Как же мне научиться, чтобы стать кудесником-хэкером? . Достаточно забавно, что, насколько можно видеть, нет никаких ЧАВО (FAQs) или Web-документов, содержащих ответ...лет в тюрьне, когда обнаружите, что не настолько ловки, насколько полагали. И это все, что я намерен **сказать** о крэкерах. Хэкерский подход Хэкеры решают проблемы и строят вещи, они верят в свободу и в...
Words matched: 5
- [ТУСОВКА: Поездка в горы II. Возвращение.](#)
...(наш разговор звучал довольно странно, т.к я не хотел показать свой странный настрой, а она не могла мне **точно** ответить о том, едет ли она) . Вот! В сети я почти никого не видел, был там недолго, зато на Инлайне...ларочки. мы помахали им и не останавливаясь шли дальше. Они отчаянно нам махали руками, что-то пытаюсь **сказать**. После чего, Дюша побежал сквозь глубокий снег и через речку, чтобы **сказать**, что доски остались в...
Words matched: 4
- [Блицкриг](#)
...достоверности Блицкрига вы слышаны, наверное, даже более меня. Реальные сражения и территории, с **точно**стью до каждого квадратного миллиметра проработанные модели боевой техники, соответствующее историческим...присутствует. Пехота еле-еле взбирается на возвышенность, зато резко спускается с них, тоже можно **сказать** и про технику. Более того, огонь с естественных возвышений много более эффективен, нежели стрельба на...
Words matched: 3
- [Исск-Куль](#)
...Исск-Куль по своей уникальности и по своей уязвимости не имеет аналогов, поскольку последнее связано с **бесточно**стью его бассейна и ограниченной экологической устойчивостью естественных комплексов...Однако интенсивное освоение богатейших природных ресурсов Принисскуля не могло не **сказать**ся на состоянии уникальной природы озера. В нем происходят необратимые процессы нарушения с далеко...
Words matched: 2
- [ТУСОВКА: Наш вариант празднования дня подруг защитников отечества...](#)
... нас впечатлили какие-то роллеры, попавшие к нам на глаза, депающие Соул. Мы облизнулись и уже почти **точно** знали, что мы делаем в субботу. Узнав накануне, что курсов не будет, мы обрадовались и чтобы хорошо...размяться и поехала на асфальт с гранита. Мы же расставили фишки, поздоровались с местными скейтбордерами. Не **сказать**, чтобы они были особо навороченными, но по-моему они смотрели на нас оценивающе. Джаста, кстати к...
Words matched: 2
- [База выпускников УК АФМШЛ №61 Якира](#)
...26 Мар 23:02:58 Товарищи! Сам являюсь выпускником нашей родной школы. Убежден, что лучше ее в Кыргызстане **точно** нет. И не только по физике/математике. Но это прелюдия. Сегодня вот обнаружил, что только на бесплатном...и хочу воспользоваться возможностью передать большой привет всем выпускникам 10 А класса 1988 года, а также **сказать** большое спасибо Хохловой Ларисе Сергеевне. Пандрей касьянов 17 Mar 00:45:00 March 17...
Words matched: 2

Search time 2 seconds; Base size: 3786 records
Поиск ведется по содержимому сайтов хостинга TO.KG и GRATIS.KG

© APCreations 2003
© Logo by B.J.
[Похилько Андрей](#)

Рис. 14. Результаты поиска на apc.gratis.kg

4.2. Поисковый сервер Kyrgyzstan On-Line

Поисковый сервер, базирующийся на Kyrgyzstan On-Line, является главной базой для тестирования всех версий ИПС, и единственным сервером, на котором к ИПС установлены дополнительные модули. К тому же, объемы хостинга содержат несколько тысяч документов, что позволяет тестировать сервер на серьезных объемах данных (см. рис. 16).

Положительным моментом является то, что сервер был заявлен на участие в конкурсе Infonet-2003 в номинации «Веб-сайты и личные страницы в Интернет» и выиграл эту номинацию, получив главный приз. [10]

Анализ навигации сайтов

<http://ilim.online.kg/>

Количество документов: 32
Количество связей: 125
Максимальный путь от документа к документу: 9
Оценка навигации: 20.8 %

[Показать трехмерную модель](#)

[Показать длиннейшие пути](#)

[Показать невозможные переходы](#)

Рис. 15. Пример анализа навигации сайта

Для ИПС на сервере Online характерен большой объем БД – порядка 150 МБ. По статистике, полезный объем данных, проходящих сквозь фильтр и попадающих в индекс, составляет всего около 30% от общего размера документов.

Также статистика и графики работы ИПС подтвердили предположения, изложенные в п. 3.3.4.4. Система зарекомендовала себя как защищенная от сбоев. Единственной проблемой может быть внезапное отключение питания, при котором происходит сбой СУБД и повреждаются хранилища.

```

1. http://www.online.kg/en/catalog/lifestyle/4/ : 200
new 1.431: make index 1
0.0002: md5
0.2093: новых ссылок: 10/117
s+ Назначено: 29.06.2004 [20:49]
0.0046: make index 2
0.2442: Words:178/64/51
0.0002: 64: 36%
0.0012: Размер: 21627 / 904

2. http://www.online.kg/ru/go/230/ : 302

3. http://www.online.kg/ru/go/228/ : 302

4. http://www.online.kg/ru/go/236/ : 302

5. http://www.online.kg/ru/go/38/ : 302

6. http://www.online.kg/ru/catalog/international/2/ : 200
new 1.6225: make index 1
0.0002: md5
0.189: новых ссылок: 6/113
s+ Назначено: 29.06.2004 [20:49]
0.0048: make index 2
0.3247: Words:280/116/86
0.0001: 116: 41%
0.0017: Размер: 21444 / 1599

```

Рис. 16. Процесс индексации

Поиск ведется на должном уровне, точность поиска не вызывает сомнений, полнота так же ограничивается лишь временем, через которое сервер успевает проиндексировать все документы (см. рис. 17).

APC SEARCH ENGINE Запрос : Искать в найденном

Найдено документов: 3

1. <http://999music.online.kg/visocky.htm>
... горани Про козла отпущения Расстрел горного эха Рядовой борисов Случай на шахте Солдат группы центр **Татуировка** Я верю в друзей Я не люблю Лирическая Am F E Здесь лапы у елей дрожат на весу, E7 Am Здесь птицы щебечут тревожно. Em7-5 A7 Dm Живешь в заколдованном диком лесу, F E7+3 Откуда уйти невозможно. Am H7 Пусть...
[<http://999music.online.kg/visocky.htm>]; 83.6KB; Релевантность: 3; Индекс цитирования: 2; Последняя проверка: 30.06.2004
2. <http://puzzle.online.kg/test/>
... начала запрещать вам брать ноутбук в постель. ваших детей зовут Яху, Ранблер или Тёна. у вас есть **татуировка**, гласящая "This body best viewed with Internet Explorer 4.0 or higher". попал в ДТП, вы инстинктивно ищете кнопку "Back". Вы представляете свою жену как www.mylady.home.wife. Ваши домашние...
[<http://puzzle.online.kg/test/>]; 32.4KB; Релевантность: 1; Индекс цитирования: 17; Последняя проверка: 30.06.2004
3. <http://999music.online.kg/zoo.htm>
... с мыслью: G A не последний ли это день? Dm Ты чувствуешь себя, как будто у тебя E На спине **татуировка** - мишень. И ты задаешь себе старый вопрос: Ну и как будет дальше жить? И ты сам себе отвечаешь: Все это глупости, их нужно забыть . Am C G Каждый день это - меткий выстрел Em Это выстрел в спину, это выстрел в...
[<http://999music.online.kg/zoo.htm>]; 19.5KB; Релевантность: 1; Индекс цитирования: 2; Последняя проверка: 30.06.2004

Time: 0.046

[О системе](#) | [Трехмерные карты сайтов](#) | [Анализ навигации](#)


 © APCreations 2003

Рис. 17. Результаты поиска

Дополнительные модули анализа навигации (см. рис. 15) и трехмерного моделирования (см. рис. 18) так же успешно работают. Единственное – обнаружилось ограничение на размер набора данных, по которому может быть произведен анализ и моделирование, так как аппаратное обеспечение не позволяет производить анализ таких больших массивов информации.

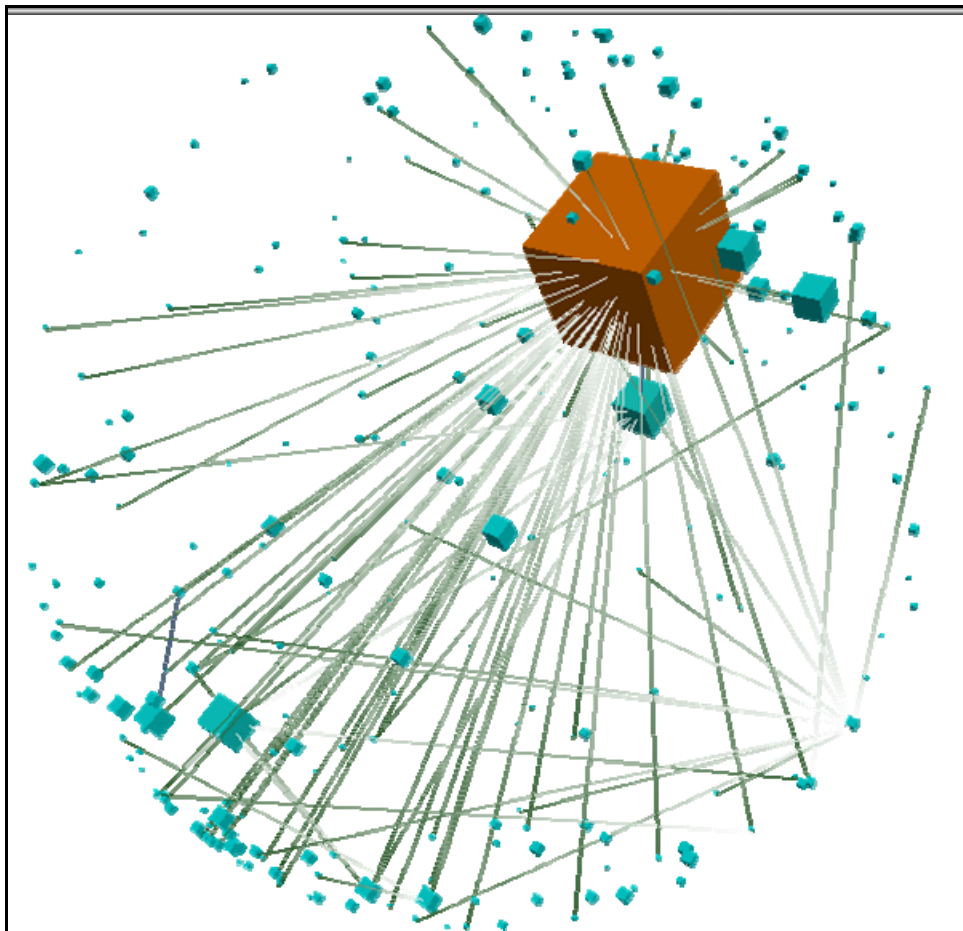


Рис. 18. Трехмерная модель сайтов

4.3. Студия NEW

Замечательным подтверждением действенности ИПС и соответствия ее требованиям реальности было предложение от студии Веб-дизайна «NEW» об использовании системы в специальном варианте для организации поиска на изготавливаемых ею сайтах (см. рис. 19). [11]

Для студии NEW были разработаны две версии поиска, сначала ТХТ2, а затем усовершенствованная следующим витком ЖЦ - ТХТ3 . Последняя имеет два варианта: основанная на предварительной индексации, и система прямого поиска по СУБД сайта.



Рис. 19. Поиск по сайту «Гражданское общество против коррупции»

4.4. Поиск для пользовательских сайтов

Еще одной реализацией ИПС, так же версии на текстовой БД явился поиск для пользовательских сайтов. Эта версия имеет программный интерфейс, позволяющий пользователю организовать поиск на своем сайте без необходимости обладать практическими навыками программиста поисковых систем. Достаточно просто начать использовать заранее подготовленные системой программы и своевременно индексировать свой ресурс.

4.5. Логотип и название семейства ИПС

В качестве названия для разработанного семейства ИПС выбрано «WEB Spider Family». При этом поисковые системы по отдельности называются APC Search Engine, или сокращенно APCSE, это объясняется традицией автора добавлять свое сетевое прозвище (никнейм) в начале названий всех своих разработок. В качестве логотипа избран паук (по аналогии с одним из главных элементов системы), сидящий на символе базы

данных с подписью WWW, которая определяет его направленность в область Интернет. Озадаченный вид паучка олицетворяет непростую задачу поиска среди огромных пространств поиска во всемирной паутине. Логотип (см. рис.20) нарисован в редакторе векторной графики XaraX, и используется на последних версиях ИПС.

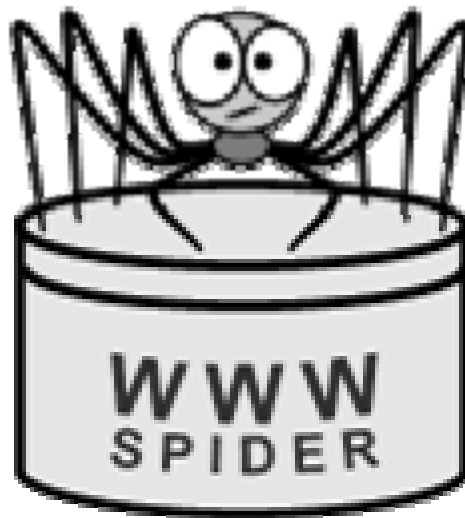


Рис. 20. Логотип семейства ИПС

5. БУДУЩЕЕ СИСТЕМЫ

5.1. Недостатки разработанной ИПС

Однако при всех успехах системы у нее достаточно недостатков, которые проявили себя на этапе внедрения и тестирования.

В первую очередь, это предел быстродействия СУБД MySQL. На тестировании по зоне .kg обнаружилось, что, если в таблице индекса порядка 7-10 млн. (!) записей, то время, затрачиваемое на поиск, начинает выходить за рамки допустимого.

Во-вторых, существенно не хватает функциональности в обработке морфологии, и использованная система RuMog не покрывает даже 30% словоформ русского языка.

В третьих, вывод информации недостаточно соответствует требованиям, а все дополнительные модули требуют доработки и финализации.

5.2. Шаг к распределенной обработке информации

Итак, основные усилия в будущем следует сосредоточить на ускорении индексации и поиска, оптимизации работы с БД. Одним из таких шагов должно стать применение *распределенной* индексации и поиска.

Принцип распределенной обработки информации является одним из ключевых в современных ИПС. Системы распределенной обработки выигрывают в соотношении цена/производительность у супер-ЭВМ, но их сложнее программировать. Большинство современных ИПС так или иначе используют распределенную обработку информации.

На данный момент автор уже ведет разработку протокола взаимодействия в сети распределенного поиска с проектным названием DSIMP (Distributed System Internal Messaging Protocol) и эксперименты с взаимодействием машин по сети.

5.3. Оптимизация запросов и структуры СУБД

Не таким радикальным шагом, как переход к распределенной системе, но, тем не менее, существенно ускоряющим работу ИПС улучшением будет дальнейшая оптимизация структуры БД и запросов к ней.

Нынешняя структура БД выполнена с хорошими характеристиками быстродействия, но есть ряд моментов, которые могут потребовать пересмотра, например, хранение полных текстов документов и других объемных записей, введение дополнительных кэширующих элементов для дальнейшего ускорения работы и другие.

5.4. Переход на другую СУБД

Как вариант решительного ускорения ИПС рассматривается переход на СУБД Oracle, но есть проблемы с тем, чтобы найти это программное обеспечение, и это потребует привязки к конкретной машине, на которой установлена эта СУБД.

Однако, ожидания от улучшений по скорости оправдывают этот переход. Вместе с тем, система потребует кардинальной адаптации к новой СУБД, но примененные принципы единого шлюза для запросов и поддержка стандартов SQL должны максимально смягчить этот процесс.

5.5. Улучшения морфологических функций

Одним из самых ощутимых улучшений должно стать применение иных и улучшение старых подходов к реализации морфологического поиска. Пока не планируется переход на сложные алгоритмы морфологического анализа, но опыт в этой области тщательно изучается.

В ближайших планах применение наряду с системой RuMog генератора словоформ для двух языков на основе словарей проекта Ispell или даже отказ от системы RuMog вообще.

5.6. Развитие дополнительных модулей

Также перспективным является развитие трехмерного моделирования сайтов и анализа навигации. В данный момент это самые новые модули ИПС и они требуют очень больших затрат по времени на усовершенствование. Ведутся консультации и исследования в области теории графов и стереометрической тригонометрии, на которых основывается работа дополнительных модулей.

ЗАКЛЮЧЕНИЕ

Разработанная поисковая система улучшит и облегчит поиск информации на сервере Kyrgyzstan On-Line, и, при должном развитии, сможет даже осуществлять региональный поиск по кыргызскому сегменту Интернет.

В результате выполнения работы глубоко изучены принципы функционирования полнотекстовых ИПС и систем поиска информации в целом. Произведено проектирование и осуществлена программная реализация полнотекстовой поисковой системы. Система внедрена на нескольких серверах и прошла полное тестирование. Выявлены пределы возможностей СУБД MySQL и получены интересные фактические данные о работе ИПС и коллекциях гипертекстовых документов.

Намечены совершенно новые перспективные направления, такие, как анализ и оценка гиперссылочной навигации и трехмерное моделирование систем связанных документов, которые основываются на данных, полученных ИПС в процессе индексации. Также выявлены недостатки разработки и указаны возможные варианты их устранения.

ГЛОССАРИЙ

Внетекстовые критерии (off-page, внедокументные) – критерии ранжирования документов в поисковых системах, учитывающие факторы, не содержащиеся в тексте самого документа и не извлекаемые оттуда никаким образом.

Дубликаты (duplicates) – разные документы с идентичным, с точки зрения пользователя, содержанием; приблизительные дубликаты (near duplicates, почти-дубликаты), в отличие от точных дубликатов, содержат незначительные отличия.

Иллюзия свежести – эффект кажущейся свежести, достигаемый поисковыми системами в интернете за счет более регулярного обхода тех документов, которые чаще находятся пользователями.

Инвертированный файл (inverted file, инверсный файл, инвертированный индекс, инвертированный список) – индекс поисковой системы, в котором перечислены слова коллекции документов, а для каждого слова перечислены все места, в которых оно встретилось.

Индекс цитирования (citation index) – число упоминаний (цитирований) научной статьи, в традиционной библиографической науке рассчитывается за промежутки времени, например, за год.

Индексирование (indexing, индексация) – процесс составления или приписывания указателя (индекса) – служебной структуры данных, необходимой для последующего поиска.

Информационный поиск (Information Retrieval, IR) – поиск неструктурированной информации, единицей представления которой является документ произвольных форматов. Предметом поиска выступает информационная потребность пользователя, неформально выраженная в поисковом запросе. И критерий поиска, и его результаты недетерминированы. Этими признаками информационный поиск отличается от «поиска данных», который оперирует набором формально заданных предикатов, имеет дело со структурированной информацией и чей результат всегда детерминирован. Теория информационного поиска изучает все составляющие процесса поиска, а именно, предварительную обработку текста (индексирование), обработку и исполнение запроса, ранжирование, пользовательский интерфейс и обратную связь.

Латентно-семантическое индексирование – запатентованный алгоритм поиска по смыслу, идентичный факторному анализу. Основан на сингулярном разложении матрицы связи слов с документами.

Лемматизация (lemmatization, нормализация) – приведение формы слова к словарному виду, то есть лемме.

Обратная встречаемость в документах (inverted document frequency, IDF, обратная частота в документах, обратная документная частота) – показатель поисковой ценности слова (его различительной силы); обратная говорят, потому что при вычислении этого показателя в знаменателе дроби обычно стоит число документов, содержащих данное слово.

Обратная связь – отклик пользователей на результат поиска, их суждения о релевантности найденных документов, зафиксированные поисковой системой и используемые, например, для итеративной модификации запроса. Следует отличать от псевдо-обратной связи – техники модификации запроса, в которой несколько первых найденных документов автоматически считаются релевантными.

Основа – часть слова, общая для набора его словообразовательных и словоизменительных (чаще) форм.

Поиск по смыслу – алгоритм информационного поиска, способный находить документы, не содержащие слов запроса.

Поиск похожих документов (similar document search) – задача информационного поиска, в которой в качестве запроса выступает сам документ и необходимо найти документы, максимально напоминающие данный.

Поисковая система (search engine, SE, информационно-поисковая система, ИПС, поисковая машина, машина поиска, «поисковик», «искалка») – программа, предназначенная для поиска информации, обычно текстовых документов.

Поисковое предписание (query, запрос) – обычно строка текста.

Полисемия (polysemy, homography, многозначность, омография, омонимия) – наличие нескольких значений у одного и того же слова.

Полнота (recall, охват) – доля релевантного материала, заключенного в ответе поисковой системы, по отношению ко всему релевантному материалу в коллекции.

Прюнинг (pruning) – отсечение заведомо нерелевантных документов при поиске с целью ускорения выполнения запроса.

Прямой поиск – поиск непосредственно по тексту документов, без предварительной обработки (без индексирования).

Различительная сила слова (term specificity, term discriminating power, контрастность, различительная сила) – степень ширины или узости слова. Слишком широкие термины в поиске приносят слишком много информации, при это существенная часть ее бесполезна. Слишком узкие термины помогают найти слишком мало документов, хотя и более точных.

Регулярное выражение (regular expression, pattern, «шаблон», реже «графарет», «маска») – способ записи поискового предписания, позволяющий

определять пожелания к искомому слову, его возможные написания, ошибки и т.д. В широком смысле – язык, позволяющий задавать запросы неограниченной сложности.

Релевантность (relevance, relevancy) – соответствие документа запросу.

Сигнатура (signature, подпись) – множество хеш-значений слов некоторого блока текста. При поиске по методу сигнатур все сигнатуры всех блоков коллекции просматриваются последовательно в поисках совпадений с хеш-значениями слов запроса.

Словоизменение (inflection) – образование формы определенного грамматического значения, обычно обязательного в данном грамматическом контексте, принадлежащей к фиксированному набору форм (парадигме), характерного для слов данного типа. В отличие от словообразования никогда не приводит к смене типа и порождает предсказуемое значение. Словоизменение имен называют склонением (declension), а глаголов – спряжением (conjugation).

Словообразование (derivation) – образование слова или основы из другого слова или основы. Чаще приводит к смене типа и к образованию слов, имеющих идиосинкразическое значение.

Спам поисковых систем (spam, спамдексинг, накрутка поисковых систем) – попытка воздействовать на результат информационного поиска со стороны авторов документов.

Стоп-слова (stop-words) – те союзы, предлоги и другие частотные слова, которые данная поисковая система исключила из процесса индексирования и поиска для повышения своей производительности и/или точности поиска.

Суффиксные деревья, суффиксные массивы (suffix trees, suffix arrays, PAT-arrays) – индекс, основанный на представлении всех значимых суффиксов текста в структуре данных, известной как бор (trie). Суффиксом в этом индексе называю любую «подстроку», начинающуюся с некоторой позиции текста (текст рассматривается как одна непрерывная строка) и продолжающуюся до его конца. В реальных приложениях длина суффиксов ограничена, а индексируются только значимые позиции – например, начала слов. Этот индекс позволяет выполнять более сложные запросы, чем индекс, построенный на инвертированных файлах.

Токенизация (tokenization, lexical analysis, графематический анализ, лексический анализ) – выделение в тексте слов, чисел, и иных токенов, в том числе, например, нахождение границ предложений.

Точность (precision) - доля релевантного материала в ответе поисковой системы.

Хеш-значение (hash-value) – значение хеш-функции (hash-function), преобразующей данные произвольной длины (обычно, строчку) в число фиксированного порядка.

Частота (слова) в документах (document frequency, встречаемость в документах, документная частота) – число документов в коллекции, содержащих данное слово.

Частота термина (term frequency, TF) – частота употреблений слова в документе.

Эксперт – специалист в предметной области, выносящий заключение о релевантности документа, найденного поисковой системой.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) The History of Yahoo! - How It All Started...
(<http://docs.yahoo.com/info/misc/history.html>)
- 2) Архитектура мета-поисковых систем
(<http://tuit.uzsci.net/libanta/internet/search/metaping.html>)
- 3) Как работают системы полнотекстового поиска
(<http://www.company.yandex.ru/articles/article10.html>)
- 4) Стемка - морфологический анализ для небольших поисковых систем
(<http://www.samag.ru/ru/articles/>)
- 5) Google Architecture
(<http://www.googleblog.ca/archives/000018.html>)
- 6) RiSearch project
(http://www.risearch.org/eng/risearch_php/index.html)
- 7) VRML Standard
(<http://xml.coverpages.org/vrmlXML9807.html>)
- 8) The Development of the VRML 97 International Standard
(<http://3dgraphics.about.com/library/weekly/aa052598.htm>)
- 9) Опыт применения методологий RAD и DATARUN в конкретных проектах
(<http://zeus.sai.msu.ru:7000/database/kbd97/11.shtml>)
- 10) Результаты конкурса в номинации "Web-сайты и личные страницы в Интернет"
(<http://infonet.kaf-i.kg/page.php?part=students&year=2003&mode=results&nom=1>)
- 11) NEW Web Design Studio: Портфолио
(<http://www.studionew.com/ru/portfolio/>)

ПРИЛОЖЕНИЯ

Приложение 1

Характеристики зарубежных поисковых систем

	Altavista	Excite	HotBot	InfoSeek	Lycos	OpenText	WebCrawler
Тип	Полнотекстовая	Полнотекстовая	Полнотекстовая	Полнотекстовая	Абстрактная	Полнотекстовая	Полнотекстовая
Размер	30 миллионов	55 миллионов	54 миллиона	20-50 миллионов	20-25 миллионов	5 миллионов	2 миллиона
Период обновления	от 1 дня до 3 месяцев	1 - 3 недели	не позднее 3 недель	от минут до месяца	ежемесячное обновление	1 - 4 недели	еженедельное обновление
Дата индексирования документа	Да	Нет	Да	Нет	Нет	Нет	Нет
Указанные (submitted) страницы	1 день	1 неделя	3 недели	1 месяц	1 месяц	2 - 4 недели	2 - 4 недели
Неуказанные (non-submitted) страницы	1 - 3 месяца	3 недели	3 недели	1 месяц	1 месяц	2 4 недели	2 4 недели
Поддержка фреймов	Нет	Да	Нет	Да	Да	Нет	Нет
Поддержка ImageMap	Да	Нет	Нет	Да	Да	Нет	Да
Защищенные паролями директории и сервера	Нет	Да	Нет	Да	Да	Нет	Нет
Частота появления ссылок	Нет	Нет	Да	Нет	Да	Нет	Да
"Обучаемость"	Да	Нет	Да	Да	Нет	Нет	Нет
Контроль индексации	robots.txt	robots.txt (в будущем и метаданные)	И то, и другое	robots.txt	robots.txt	robots.txt	И то, и другое
Перенаправление (redirect)	Поддерживает	Поддерживает	-	-	-	-	Поддерживает
Стоп-слова	Да	Да	Да	Нет	Да	Нет	Нет
Влияние на алгоритм определения релевантности	Нет	-	Ключевые слова в метаданных	Нет	Нет	Нет	Частота появления ссылок
Spart-штрафы	Да	Да	Да	Да	Да	Да	Да
Поддержка META-тегов	Да	Нет	Да	Да	Да	Нет	Только NOINDEX tag
Title	Заголовок страницы или No Title	Заголовок страницы или Untitled	Заголовок страницы или URL	Заголовок страницы или первая строка документа	Заголовок страницы или первая строка документа	Первые 100 символов из документа	Заголовок страницы или URL
Description	Метатег или первые несколько строк из документа	Формируется из наиболее релевантных к запросу фраз документа	Метатег или первые несколько строк документа	Метатег или первые 200 символов после тега <body>	Метатег или экстакт из содержимого страницы	Первые 100 символов документа	Создается из содержания; обещается поддержка метатегов в будущем
Проверка статуса URL	Да	Нет	Нет	Нет	Да	Нет	Да
Удаление старых данных	Удалить содержимое и указать новый адрес	Удалить содержимое или переписать robots.txt	Переписать robots.txt	Удалить содержимое и указать новый адрес или переписать robots.txt	-	-	-

Информация приведена Calafia Consulting по состоянию на начало 1998 года.

Характеристики российских поисковых систем

	Russian Express	TELA поиск	Rambler	Yandex	Апорт Поиск
Тип	Полнотекстовая	Полнотекстовая	Полнотекстовая	Полнотекстовая	Полнотекстовая
Размер	500.000	140.094	2.500.000	2.000.000	2.600.000
Период обновления	20 дней	3-4 недели	1 раз в неделю	перманентно	раз в сутки (от 10 до 40 тысяч документов)
Дата обновления	Нет, в проекте да	Да	Да, при расширенной выдаче результатов	Да	Да
Неуказанные (non-submitted) страницы	20 дней	-	до 3 месяцев	в зависимости от популярности документов	лимитируется скоростью обновления индекса
Глубина индексирования	5.000 документов на глубину 150	20 документов	не ограничена	не ограничена	не ограничена
Поддержка фреймов	Да	Да	Да	Да	Да
Поддержка ImageMap	Да	Да	Да	Да	Да
Учет популярности документа при реиндексировании	Нет (в проекте - да)	Нет	Нет	Да	Нет
Использование robots.txt <META ROBOTS=...	Да	robots.txt - да META - нет	Да	Да	Да
Влияние на алгоритм определения релевантности	Нет	Пока нет, в проекте - META-Keywords	Нет и не будет	Пока не поддерживаются	Пока не поддерживаются
Title	пока URL	title	title или URL и относительная мера релевантности	title и URL	title
Description	META-таг Description и часть текста документа	Первые строки документа	Первые 512 байт документа исключая meta, javascript, images... Существуют еще две формы вывода описания - короткая и длинная	Выдаются первые 1024 байт текста, мера релевантности, дата создания и объем документа	Предложения, содержащие слова запроса (1, 3 или до 10)
Контроль над индексацией	Нет	Явно - нет, косвенно - указав в качестве критерия URL	Да (См. подробнее)	Пока нет	Да (См. подробнее)

Составил Андрей Аликберов, ЦИТ. Последние изменения 6 января 1998 года

Графики работы разработанной ИПС за 4185 итераций

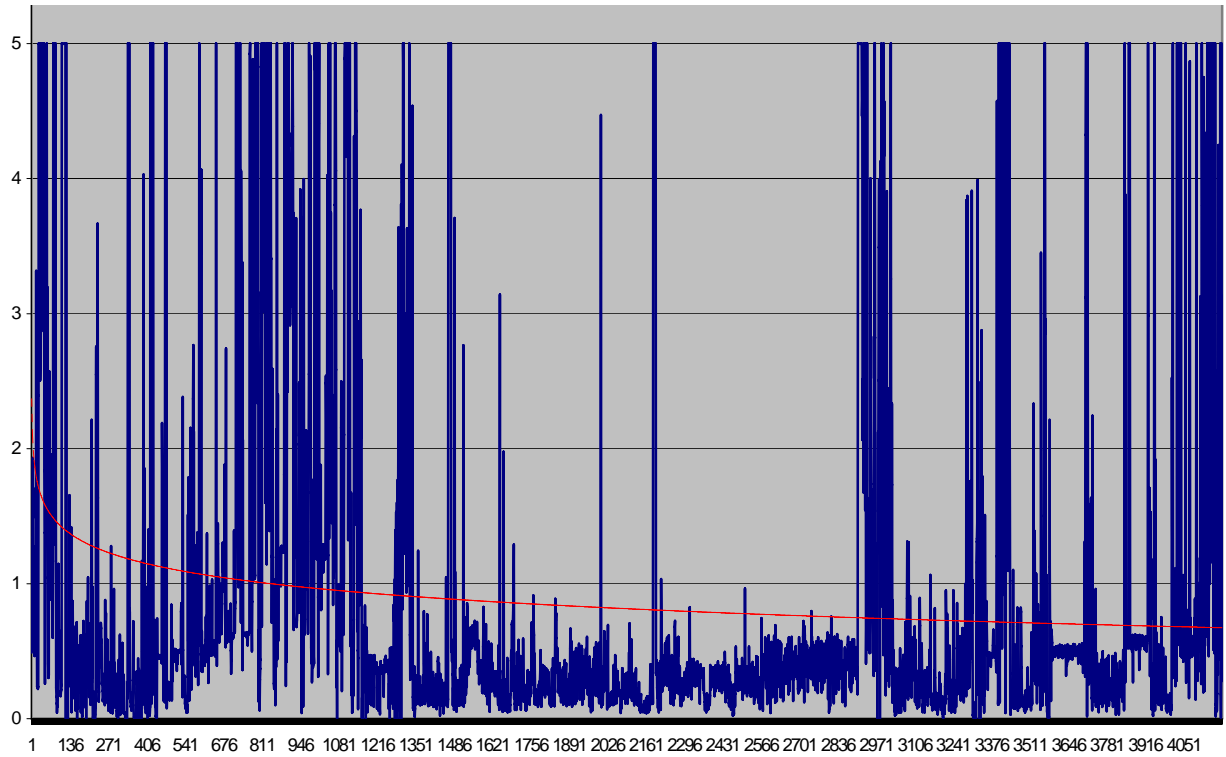


Рис. 21. Скорость индексации

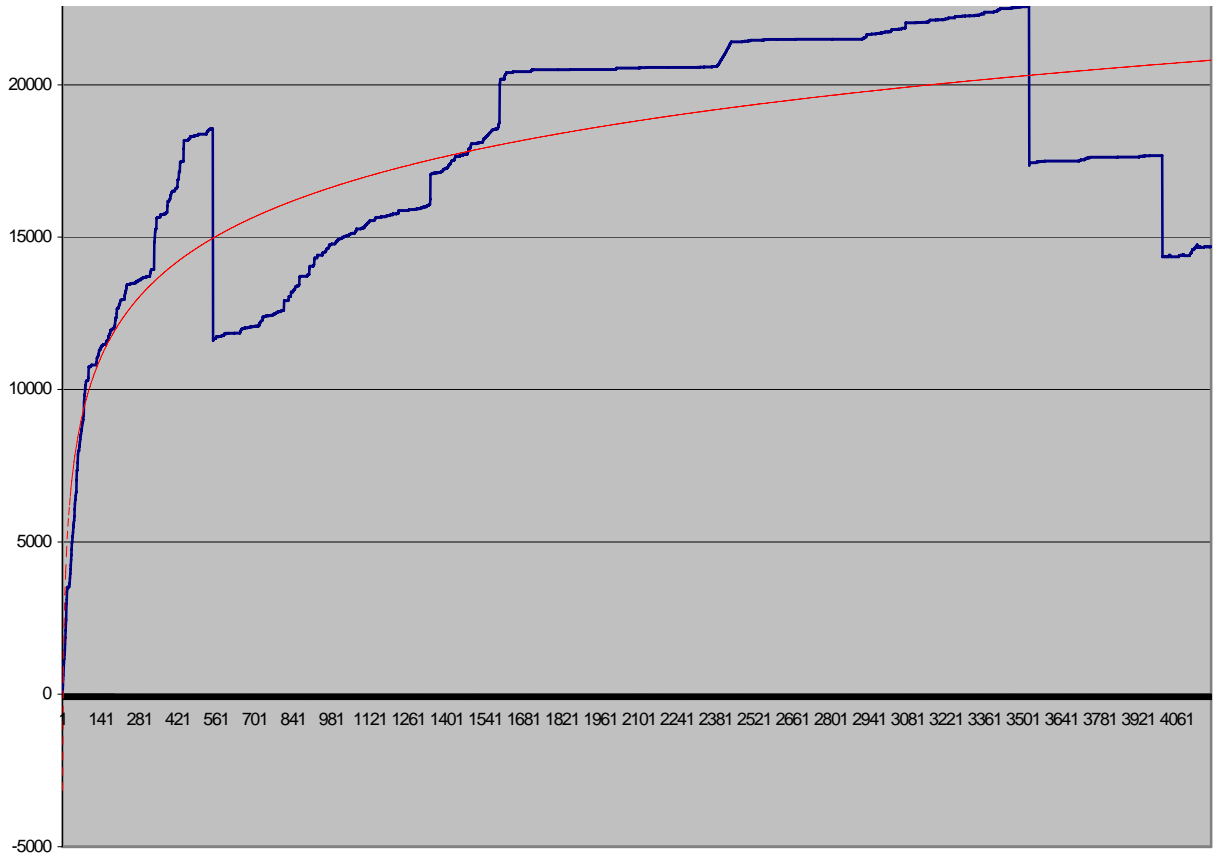


Рис. 22. Количество документов в очереди

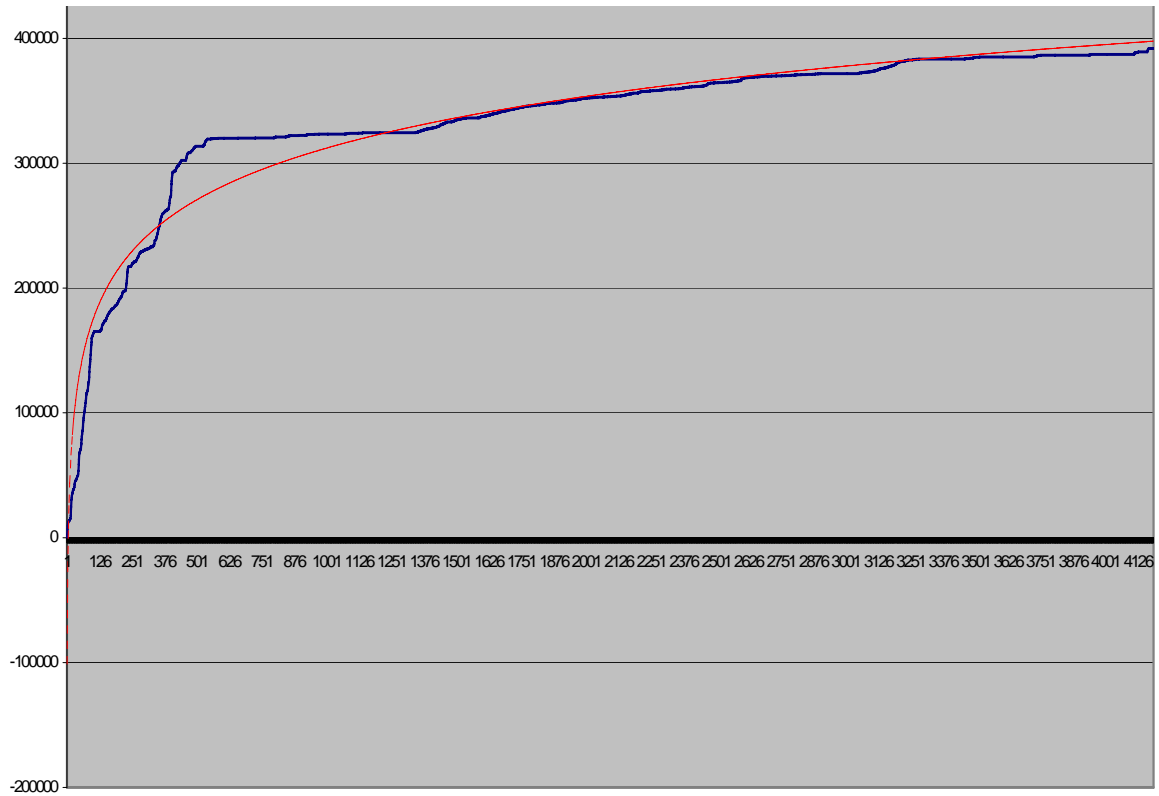


Рис. 23. Количество слов в словаре

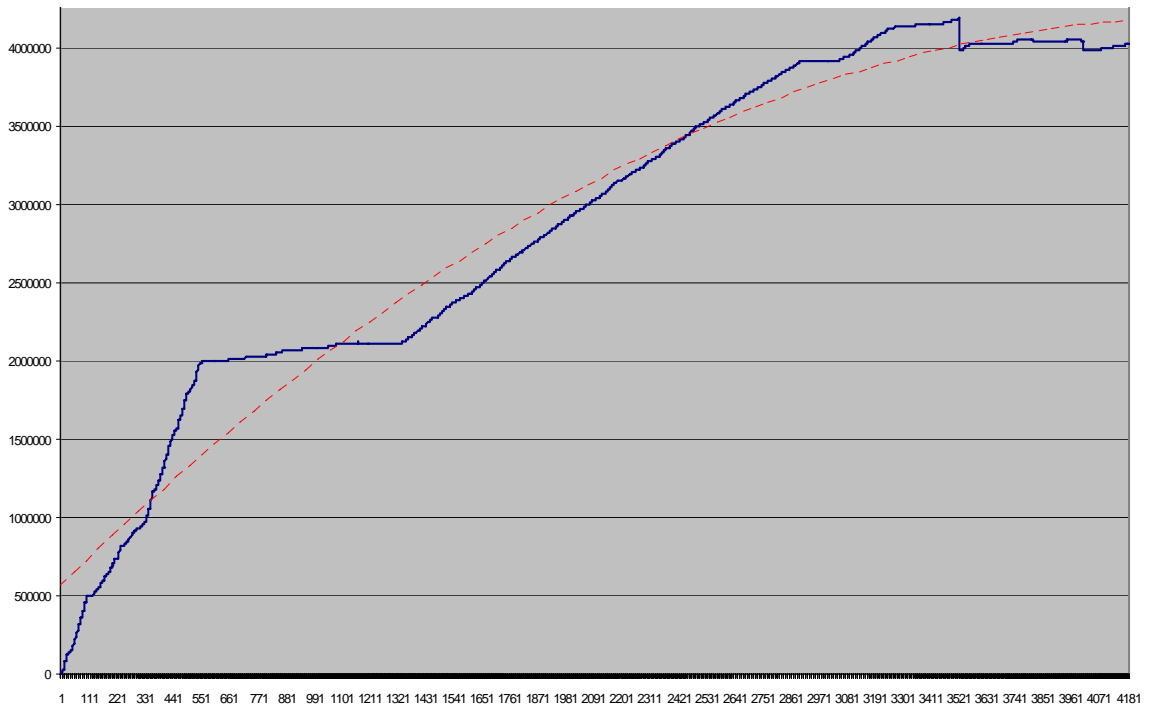


Рис. 24. Размер таблицы индекса

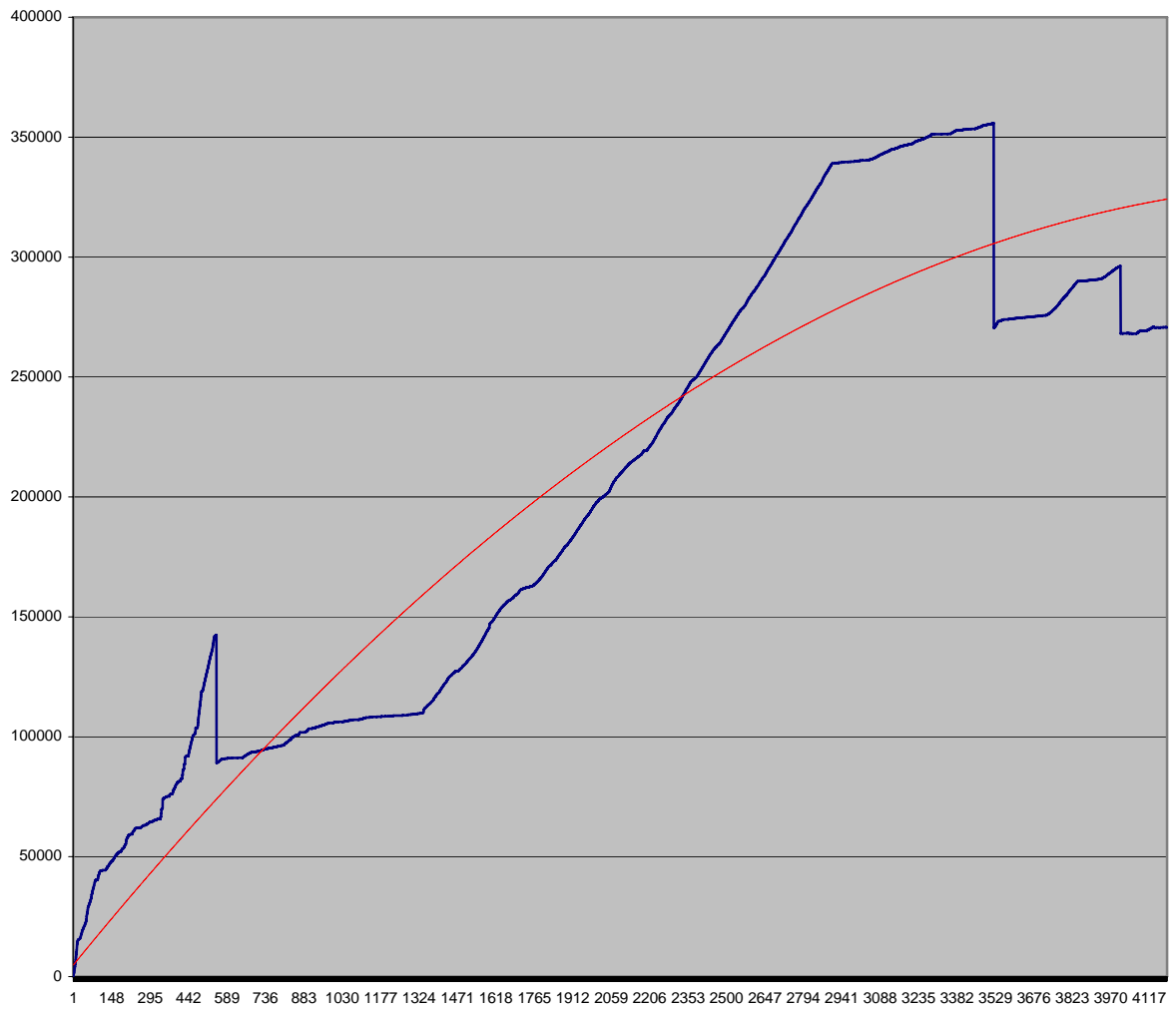


Рис. 25. Зарегистрированные ссылки между документами