**МИНИСТЕРСТВО ОБРАЗОВАНИЯ и науки РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Белгородский государственный университет**

**Старооскольский филиал**

**реферат**

**Поисковые системы интернета**

**на примере системы «Rambler»**

Выполнил: Дёменко Алексей.

студент 150 группы

специальности «Педагогика и методика начального обучения»

Проверила: Карнаухова М. В.

Старый Оскол - 2004

## Содержание

**ВВЕДЕНИЕ……………………………………………………………….……………………3**

**Язык поисковых запросов……………………………………………………....…3**

**Регистр……………………………………………………………………………………….3**

**Операторы………………………………………………………………………………….3**

**Кавычки……………………………………………………………………………………..4**

**Скобки………………………………………………………………………………………..4**

**Метасимволы…………………………………………………………………………….4**

**Применение языка запросов…………………………………………………..….4**

**Морфология……………………………………………………………………………….5**

**Стоп-слова………………………………………………………………………………...5**

**Ограничение расстояния………………………………………………………….5**

**Ненайденные слова…………………………………………………………………..5**

**Специальные операторы…………………………………………………………..5**

### Комфортный поиск…………………………………………………………………...5

#### Как включить панель ссылок…………………………………………………..6

### Как сделать, чтобы Rambler находил мои документы?…………..6

### Как управлять индексированием сайта, Использование файлов robots.txt, Роботы и файл robots.txt………………………...7

**Размещение файла robots.txt…………………………………………………8**

**Формат файла robots.txt…………………………………………………………8**

**Группы инструкций для отдельных роботов: User-agent……..…8**

**Пустые строки и комментарии………………………………………………..9**

**Использование META-тегов "Robots"……………………………………….9**

### Определение позиции сайта в результатах поиска по заданному запросу……………………………………………………………….…9

### Принципы работы поисковой машины Рамблер………………….…9

**Заключение………………………………………………………………………….….16**

##### Введение

Здесь описаны:

* Язык поисковых запросов: что и как можно написать в поисковой строке;
* Применение языка запросов: как поисковая машина Рамблера обрабатывает запрос;
* Расширенный поиск: как искать в Рамблере более эффективно, пользуясь страницей Расширенного поиска;
* Комфортный поиск: как установить специальную кнопку для поиска в Рамблере прямо на панель браузера.

А также рекомендации для владельцев сайтов:

* Как сделать, чтобы Rambler находил мои документы;
* Как управлять индексированием сайта;
* Ответы на часто задаваемые вопросы.

**Язык поисковых запросов**

Поисковый запрос может состоять из одного или нескольких слов, в нем могут присутствовать знаки препинания. Составлять простые запросы можно и не вдаваясь в тонкости языка запросов. Так, если ввести в поисковую строку несколько слов без знаков препинания и логических операторов, будут найдены документы, содержащие все эти слова (причем на ограниченном расстоянии друг от друга). Знание и правильное применение языка запросов поисковой машины поможет сделать поиск на Рамблере быстрым и эффективным.

**Регистр**

В общем случае, регистр написания поисковых слов и операторов значения не имеет, то есть дом и ДОМ, Not и nOt воспринимаются одинаково. И лишь иногда, в целях повышения качества поиска, регистр слов поискового запроса принимается во внимание.

Например, если запрос состоит из двух, трех или четырех слов, каждое из которых написано с большой буквы, то предполагается поиск по имени собственному, и автоматически производится изменение ограничения расстояния между словами запроса со значения по умолчанию на величину (n-1)\*2, где n - количество слов запроса. Это позволяет находить группу слов запроса, внутри которой есть не более одного "лишнего" слова или знака препинания, например "Баден-Баден", "А. Пушкин", "Федор Михайлович Достоевский".

**Операторы**

Запрос, состоящий из нескольких слов, может содержать операторы. Поиск операторов в документе не производится, они служат лишь инструкцией поисковой машине. Все операторы поисковой машины бинарные, то есть имеют левую и правую часть, каждая из которых также является запросом (по умолчанию состоящим из одного слова). Для изменения сферы действия операторов (группировки нескольких слов запроса в аргумент оператора) применяются скобки и кавычки.

Два запроса, соединенные оператором AND (логическое И) образуют сложный запрос, которому удовлетворяют только те документы, которые одновременно удовлетворяют обоим этим запросам. Иными словами, по запросу **'собака AND кошка'** найдутся только те документы, которые содержат и слово **'собака'**, и слово **'кошка'**.

Сложному запросу, состоящему из двух запросов, соединенных оператором OR (логическое ИЛИ) удовлетворяют все документы, удовлетворяющие хотя бы одному из этих двух запросов. По запросу **'собака OR кошка'** найдутся документы, в которых есть хотя бы одно из слов **'собака'** или **'кошка'** (либо оба эти слова вместе).

Оператор NOT (логическое И-НЕ) образует запрос, которому отвечают документы, удовлетворяющие левой части запроса и не удовлетворяющие правой. Так, результатом поиска по запросу **'собака NOT кошка'** будут все документы, в которых есть слово **'собака'** и нет слова **'кошка'**.

Если оператор явно не указан, используется оператор по умолчанию AND: находятся только документы, содержащие все слова запроса. Так, запрос **'информация технологии кредит'** будет истолкован как **'информация AND технологии AND кредит'**. На странице Расширенного поиска оператор по умолчанию можно заменить на OR (Искать слова запроса: хотя бы одно).

Каждый из операторов имеет сокращенное обозначение:

|  |  |
| --- | --- |
| **Оператор** | **Сокращенное обозначение** |
| **AND** | **&** |
| **OR** | **|** |
| **NOT** | **!** |

Запрос из нескольких слов, перемежающихся операторами, будет истолкован в соответствии с их приоритетом. Операторы AND и NOT традиционно имеют более высокий приоритет, поэтому запрос из нескольких слов при обработке сначала группируется по операторам AND и NOT, и лишь потом по операторам OR. Изменить порядок группировки можно использованием скобок.

**Кавычки**

Для поиска цитат можно использовать двойные кавычки. Слова запроса, заключенного в двойные кавычки, ищутся в документах именно в том порядке и в тех формах, в которых они встретились в запросе.

Таким образом, двойные кавычки можно использовать и просто для поиска слова в заданной форме (по умолчанию слова находятся во всех формах). Например, запросу **'самолет "заправился" посадка'** удовлетворяет документ, содержащий текст **'... самолет совершил посадку и заправился ...'**, и не удовлетворяет документ, содержащий **'.. самолет совершил посадку, чтобы заправиться ...'**.

**Скобки**

При построении запросов иногда возникает необходимость объединения слов запроса в группы, которые будут аргументами некоторого оператора. Такие группы заключаются в скобки.

Часть запроса, заключенная в скобки, сама является запросом, и на нее распространяются правила языка построения запросов. Использование скобок позволяет строить вложенные запросы и передавать их операторам в качестве аргументов, а также перекрывать приоритеты операторов, принятые по умолчанию.

Если запрос без скобок **'машина самолет | аэродром'** эквивалентен запросу **'машина AND самолет OR аэродром'** и, в соответствии с приоритетами операторов, означает "найти документы, содержащие либо слова **'машина'** и **'самолет'**, либо слово аэродром, то запрос со скобками **'машина (самолет | аэродром)'** равносилен запросу **'машина AND (самолет OR аэродром)'**, что означает "найти документы, содержащие слово **'машина'** и одно из слов **'самолет'** или **'аэродром'**".

**Метасимволы**

Рамблер пока не поддерживает поиск строк с использованием метасимволов (**'\*'**, **'?'**), которые обычно используются в значении "любая подстрока" и "произвольный одиночный символ" соответственно. Тем не менее, эти операторы зарезервированы для подобного использования в будущем.

**Применение языка запросов**

Каждый запрос, адресованный поисковой машине Рамблера, обрабатывается в соответствии с правилами языка запросов. Некоторые слова и символы трактуются как операторы языка запросов и обрабатываются специальным образом. Фактически, языком запросов описывается некая формула, которая используется при поиске - каждый из документов "сопоставляется" с ней, и результатом поиска являются только те документы, которые ей удовлетворяют.  
Например, запросу **'самолет'** удовлетворяют все документы, в которых хотя бы раз встретилось слово **'самолет'** в любой форме. Запросу, состоящему из нескольких слов, удовлетворяют документы, содержащие каждое из этих слов в любой форме (при некоторых условиях). Вопрос соответствия документа более сложному запросу определяется логикой операторов и конструкций языка запросов.

**Морфология**

По каждому слову запроса поиск ведется с учетом правил словоизменения соответствующего языка. Рамблер понимает и различает слова русского и английского языков - по умолчанию, поиск ведется по всем формам слова.

Например, при поиске по слову **'человек'** будут также найдены документы, содержащие слова **'человеку'**, **'человеком'**, **'человека'** и даже **'люди'**. Чтобы провести поиск только по одной определенной форме слова, нужно взять его в двойные кавычки или воспользоваться поиском точной фразы в расширенном поиске.

**Стоп-слова**

Некоторые слова и символы по умолчанию исключаются из запроса в связи с их малой информативностью. Это так называемые стоп-слова - самые частотные слова русского и английского языков, например, предлоги, частицы и артикли. Присутствие этих слов может замедлить поиск и негативно повлиять на полноту результатов. Есть возможность обозначить необходимость этих слов в запросе, взяв запрос в двойные кавычки или воспользовавшись поиском точной фразы в расширенном поиске.

**Ограничение расстояния**

Если запрос составлен из одного или нескольких слов без применения операторов и конструкций языка запросов, то будут найдены документы, в которых встречаются все слова запроса. При этом для каждого запроса всегда существует так называемое ограничение контекста - положительное число, по умолчанию равное расстоянию в 40 слов. Документ, в котором встретились все слова запроса, будет выдан только в том случае, если расстояние в словах между вхождениями слов запроса будет меньше этого числа. Например, по запросу **'красная армия'** будут найдены те документы, в которых слова **'красная'** и **'армия'** хотя бы один раз встретятся менее чем в 40 словах друг от друга.

Значение ограничения контекста можно изменять конструкцией **'(число, запрос)'**, где число - любое положительное число, запрос - любой корректный с точки зрения поисковой машины запрос, состоящий более чем из одного слова (очевидно, ограничение расстояния между словами в случае однословного запроса не имеет смысла). Таким образом, по запросу **'(2, красная армия)'** найдутся только те документы, в которых между словами **'красная'** и **'армия'** хотя бы раз не стоит ни одного слова (поскольку лишь в случае их непосредственного соседства разница в порядковых номерах слов меньше 2, т.е. равна 1)

**Ненайденные слова**

Если запрос состоит из нескольких слов, и при этом некоторые из них вообще не удалось найти в Интернете, то выдаются результаты поиска по частичному запросу, из которого отсутствующие в Интернете слова исключены. При этом на странице результатов поиска выдается соответствующая диагностика.

**Специальные операторы**

Рамблер позволяет искать страницы, на которых размещены счетчики Top100, TopShop, TopList, SpyLog, а также HotLog. Для того, чтобы найти в интернете все страницы, на которых размещен счетчик с заданным идентифтикатором, используйте оператор *${counter=ID}*, где counter - название счетчика (top100, topshop, toplist, spylog или hotlog), а ID - номер счетчика (идентификатор ресурса).

**Пример**: для того, чтобы найти в Интернете все страницы раздела Рамблер-Открытки (идентификатор Top100 - 193680), подайте Рамблеру запрос ${top100=193680}.

### Комфортный поиск

Для облегчения поиска в Internet можно установить на панель броузера (Netscape или Internet Explorer версии не ниже 4) специальную кнопку поиска в Rambler. Как это сделать?

Перетащить мышкой одну из ссылок на специальной страничке в поле панели ссылок.

После этого на панели появится кнопка "Искать в Rambler". В броузере должно быть разрешено выполнение JavaScript. Если во время просмотра документа выделить текст, который надо задать в качестве поискового запроса, и нажать на эту кнопку, запрос будет передан **Rambler**. Результаты поиска будут выведены в другом окне. Длина запроса ограничена 96 символами

#### Как включить панель ссылок

Если панель ссылок отключена, то включить ее можно следующим образом:

Netscape

В меню 'View' отметьте 'Show Personal Toolbar'

Internet Explorer

В меню 'View' (или 'Вид') определите 'ToolBars' ('Панели инструментов'). Затем пометьте 'Links' ('Ссылки')

### Как сделать, чтобы Rambler находил мои документы?

1. Прежде всего надо заполнить регистрационную анкету в поисковой системе Rambler. Это будет гарантией того, что роботы Рамблера узнают о сайте и скорее начнут его индексацию. Анкета находится по адресу http://www.rambler.ru/doc/add\_site\_form.shtml.
2. Автоматически роботы Rambler сканируют сайты, находящиеся в следующих доменах первого уровня:

Российская Федерация: .ru, .su  
Украина: .ua  
Белоруссия: .by  
Казахстан: .kz  
Киргизия: .kg  
Узбекистан: .uz  
Грузия: .ge

и игнорируют сайты из других доменов.

Если данный сайт находится вне названных доменов (например, в зонах *.com, .org, .net*), но существенная часть сайта содержит русскоязычные материалы или он может представлять интерес для русскоязычной аудитории Рамблера, можно отослать письмо на адрес search.support@rambler-co.ru с просьбой включить сайт в число сканируемых, либо заполнить форму обратной связи. Сотрудники Рамблера рассмотрят эту просьбу и примут решение о целесообразности такого включения.

1. Рекомендуется зарегистрировать сайт в рейтинге Top100 и расставить счетчик на всех страницах сайта. Анкета, заполняемая при регистрации в этом рейтинге, индексируется ежедневно, а специальный робот Рамблера дважды в день пополняет базу поисковой машины новыми страницами, на которых размещен счетчик. Таким образом, включение сайта в Тор100 - это самый быстрый способ попасть в результаты поиска!
2. При заполнении полей анкеты "Название сайта" и "Описание" не следует вводить в них длинные перечни ключевых слов. Эти поля все равно пока не используются для поиска. Название и описание должны быть предназначены для прочтения человеком, так как эти поля используются в наших внутренних базах данных и просматриваются редакторами.
3. Рамблер умеет извлекать гиперссылки из объектов Macromedia Flash. Если сайт имеет заставку или навигационные панели, выполненные c использованием этой технологии, Рамблер обработает их, найдет адреса всех страниц сайта и проиндексирует весь сайт. Однако, сами тексты flash-объектов не индексируются. Это решение принято потому, что большая часть таких объектов содержит элементы навигации, заставки, меню и другие фрагменты, очень важные в качестве источника гиперссылок, но малоинформативные как текст. Для сайтов, которые целиком состоят из flash-объектов, рекомендуется создать HTML-копию и зарегистрировать ее в поисковой машине.
4. Роботы Рамблера при сканировании игнорируют поля **<meta name="keywords"...>** и все другие поля **<meta...>**, кроме **<meta name="robots"...>**. Это связано с тем, что эта система старается индексировать документ таким, какой он есть (то есть таким, каким его видит пользователь). Не секрет, что зачастую создатели интернет-страниц злоупотребляют этими полями, пытаясь заставить поисковые машины находить документ по запросам, не имеющим к нему прямого отношения. Не следует также использовать невидимый текст (в котором цвет шрифта совпадает с цветом фона). Комментарии в документе роботы Рамблера тоже не сканируют, поэтому использовать их лучше по прямому назначению. Помните, что каждый комментарий увеличивает размер документа, а значит, снижает вероятность того, что документ будет просмотрен пользователем до конца.
5. Обратите внимание на заголовки и выделения в документе. Базовые понятия и ключевые для данного сайта слова целесообразно включать в следующие HTML-теги (в порядке значимости):

**<title>  
<h1>...<h4>  
<b>, <strong>, <u>**

Чем чаще слово встречается в этих полях, тем более вероятно, что поисковая система Rambler выдаст ссылку на Ваш документ ближе к началу списка результатов поиска. Конечно, использование этих тегов должно органично сочетаться с дизайном Вашего сайта.

1. С точки зрения поиска, использование фреймов в документе не приветствуется. Это не означает, что роботы не умеют сканировать фреймы. Роботы Rambler прекрасно справляются с конструкциями фреймов, однако наличие лишнего этажа ссылок (от головного навигационного фрейма к "содержательным") замедляет индексацию.

Оптимальным является включать в документы с фреймами HTML-тег **<noframes>** с текстом документа и ссылками. Разумеется, это увеличит размер документа, но будет являться актом доброй воли по отношению к пользователям текстовых браузеров (например, Lynx) и поисковым машинам.

1. Максимальный размер документа для роботов Рамблера составляет 200 килобайт. Документы большего размера усекаются до указанной величины. Впрочем, размещать в Сети документы такого размера без особой на то необходимости - все равно дурной тон; в любом случае надо ограничивать объем документа разумными рамками.
2. Роботы Рамблера обрабатывают ссылки типа **<a href=".../imagemap ...">**, однако наряду со ссылкой такого вида хорошо бы поместить в текст документа конструкцию **<map name="name">**. Это ускорит индексацию документов, указанных в *imagemap*, и облегчит доступ к документам для обычных браузеров.
3. При написании документов надо внимательно следить за соблюдением русского/латинского регистров. Часто, например вместо русской буквы 'р' используют латинскую 'p', вместо русского 'с' - латинское 'c'. Некоторые подобные ошибки индексатор исправляет, но не все. Слова с подобными опечатками теряют информативность.

Старайтесь не использовать дефисы '-' в качестве символов переноса. При этом слова разбиваются и теряют информативность; кроме того, такие переносы имеют все шансы оказаться у пользователя в середине строки. Помните, что браузер сам осуществляет представление документа согласно текущим установкам каждого конкретного пользователя.

1. Часто изменяющиеся (динамические) документы рекомендуется исключить из списка индексируемых, т. к. актуальность этих документов быстро теряется. Осуществить это можно с помощью стандартного для HTTP механизма - посредством файла robots.txt в головной директории Вашего сайта или HTML-тега **<meta name="robots" ...>**.

Части документа, не требующие, по Вашему мнению, индексации, можно отделять в документе с помощью тегов **<noindex> ... </noindex>**. Из частей документа, размеченных этими тегами, также не будут выделены ссылки для дальнейшего обхода.

1. При задании перекрестных ссылок в документе будьте предельно внимательны, проверьте работоспособность каждой ссылки, иначе роботы (и пользователи!) не смогут добраться до некоторых документов.

Следует также иметь в виду, что с точки зрения HTML записи типа:

**<a href="directory" ...>** и  
**<a href="directory/" ...>** ("слэш" в конце *href*)

являются разными ссылками. Обычно при запросе по первой ссылке робот получит редирект на вторую, а значит извлечет сам документ при обращении к серверу только на следующем проходе. Тем самым замедлится индексация сайта.

1. Необходимо относится к планированию и размещению сайта серьезно, чтобы впоследствии не пришлось забрасывать администраторов поисковых систем письмами с просьбой переиндексировать сайт в связи с его переносом или полным изменением структуры. Поисковые машины - вещь достаточно инерционная, и переиндексация не будет мгновенной.

### Как управлять индексированием сайта

**Использование файлов robots.txt**

**Роботы и файл robots.txt**

Рамблер, как и другие поисковые машины, для поиска и индексации интернет-ресурсов использует программу-робот. Робот скачивает документы, выставленные в Интернет, находит в них ссылки на другие документы, скачивает вновь найденные документы и находят в них ссылки, и так далее, пока не обойдет весь интересующий его участок Сети. Называется этот робот **StackRambler**.

Когда робот-индексатор поисковой машины приходит на web-сайт (к примеру, на http://www.rambler.ru/), он прежде всего проверяет, нет ли в корневом каталоге сайта служебного файла **robots.txt** (в нашем примере - http://www.rambler.ru/robots.txt).

Если робот обнаруживает этот документ, все дальнейшие действия по индексированию сайта осуществляются в соответствии с указаниями **robots.txt**. Можно запретить доступ к определенным каталогам и/или файлам своего сайта любым роботам-индексаторам или же роботам конкретной поисковой системы.

Правда, инструкциям файла **robots.txt** (как и meta-тегов **Robots**, см. ниже) следуют только так называемые "вежливые" роботы - к числу которых робот-индексатор Рамблера, разумеется, относится.

**Размещение файла robots.txt**

Робот ищет **robots.txt** только в корневом каталоге сервера. Под именем сервера здесь понимаются доменное имя и, если он есть, порт. Размещать на сайте несколько файлов **robots.txt**, размещать **robots.txt** в подкаталогах (в том числе подкаталогах пользователей типа *www.hostsite.ru/~user1/*) бессмысленно: "лишние" файлы просто не будут учтены роботом. Таким образом, вся информация о запретах на индексирование подкаталогов сайта должна быть собрана в едином файле **robots.txt** в "корне" сайта. Имя **robots.txt** должно быть набрано строчными (маленькими) буквами, поскольку имена интернет-ресурсов (URI) чувствительны к регистру. Ниже приведены примеры правильных и неправильных размещений **robots.txt**.

**Правильные:**  
http://www.w3.org/robots.txt  
http://w3.org/robots.txt  
http://www.w3.org:80/robots.txt  
(В данном случае все эти три ссылки ведут на один и тот же файл.)  
  
**Неправильные:**  
http://www.yoursite.ru/publick/robots.txt  
http://www.yoursite.ru/~you/robots.txt  
http://www.yoursite.ru/Robots.txt  
http://www.yoursite.ru/ROBOTS.TXT

**Формат файла robots.txt**

**Пример**

Следующий простой файл **robots.txt** запрещает индексацию всех страниц сайта всем роботам, кроме робота Рамблера, которому, наоборот, разрешена индексация всех страниц сайта.

# Инструкции для всех роботов  
User-agent:   \*  
Disallow: /  
  
# Инструкции для робота Рамблера  
User-agent: StackRambler  
Disallow:

**Группы инструкций для отдельных роботов: User-agent**

Любой файл **robots.txt** состоит из групп инструкций. Каждая из них начинается со строки User-agent, указывающей, к каким роботам относятся следующие за ней инструкции Disallow.

Для каждого робота пишется своя группа инструкций. Это означает, что робот может быть упомянут только в одной строке User-agent, и в каждой строке User-agent может быть упомянут только один робот.

Исключение составляет строка User-agent:   \*. Она означает, что следующие за ней Disallow относятся ко всем роботам, кроме тех, для которых есть свои строки User-agent.

**Инструкции: Disallow**

В каждой группе, вводимой строкой User-agent, должна быть хотя бы одна инструкция Disallow. Количество инструкций Disallow не ограничено.

Строка "Disallow: /dir" запрещает посещение всех страниц сервера, полное имя которых (от корня сервера) начинается с "/dir". Например: "/dir.html", "/dir/index.html", "/directory.html".

Чтобы запрещать посещение именно каталога "/dir", инструкция должна иметь вид: "Disallow: /dir/". Для того, чтобы инструкция что-либо запрещала, указанный в ней путь должен начинаться с "/". Соответственно, инструкция "Disallow:" не запрещает ничего, то есть все разрешает.

**Внимание:** точно так же и инструкции "Disallow:   \*", "Disallow: \*.doc", "Disallow: /dir/\*.doc" не запрещают ничего, поскольку файлов, имя которых начинается со звездочки или содержит ее, не существует! Использование регулярных выражений в строках Disallow, равно как и в файле **robots.txt** вообще, не предусмотрено.

К сожалению, инструкций Allow в файлах **robots.txt** не бывает. Поэтому даже если закрытых для индексирования документов очень много, все равно придется перечислять именно их, а не немногочисленные "открытые" документы. Надо продумать структуру сайта, чтобы закрытые для индексирования документы были собраны по возможности в одном месте.

**Пустые строки и комментарии**

Пустые строки допускаются между группами инструкций, вводимыми User-agent.

Инструкция Disallow учитывается, только если она подчинена какой-либо строке User-agent - то есть если выше нее есть строка User-agent.

Любой текст от знака решетки "#" до конца строки считается комментарием и игнорируется.

**Использование META-тегов "Robots"**

В отличие от файлов **robots.txt**, описывающих индексацию сайта в целом, тег <meta name="Robots" content="..."> управляет индексацией конкретной web-страницы. При этом роботам можно запретить не только индексацию самого документа, но и проход по имеющимся в нем ссылкам.

Инструкции по индексации записываются в поле content. Возможны следующие инструкции:

* NOINDEX - запрещает индексирование документа;
* NOFOLLOW - запрещает проход по ссылкам, имеющимся в документе;
* INDEX - разрешает индексирование документа;
* FOLLOW - разрешает проход по ссылкам.
* ALL - равносильно INDEX, FOLLOW
* NONE - равносильно NOINDEX, NOFOLLOW

Значение по умолчанию: <meta name="Robots" content="INDEX, FOLLOW">.

В следующем примере робот может индексировать документ, но не должен выделять из него ссылки для поиска дальнейших документов:

<META name="ROBOTS" content="index, nofollow">

Имя тега, названия и значения полей нечувствительны к регистру.

В поле content дублирование инструкций, наличие противоречивых инструкций и т.п. не допускается; в частности, значение поле content не может иметь вид "none, nofollow".

### Определение позиции сайта в результатах поиска по заданному запросу

В ходе проверки Рамблер просматривает примерно 650 первых результатов поиска по заданному запросу и ищет в них интересующий пользователя сайт. Если этот сайт найден, результат поиска содержит его позицию в общей выдаче (на странице результатов поиска Rambler'a) и ссылку на наиболее релевантную страницу сайта. Если же сайт найти не удалось, выдаются первые 15 сайтов.

### Принципы работы поисковой машины Рамблер,

*или как выжить в условиях постоянно растущего Интернета*

Интернет постоянно растет, так же как растет и число пользователей, которые обращаются с запросами к поисковым системам. Увеличение объема информации и количества запросов, в свою очередь, приводит к повышению требований к скорости работы поисковых машин, качеству поиска и наглядности представления результатов. Так, для того чтобы пользователь остался доволен результатом, на сегодняшний день поисковой системе нужно собрать, обработать, обновить, найти и отсортировать в два раза больше документов, чем год назад. А основная задача поиска как раз и состоит в том, чтобы пользователь был доволен его результатами.

Когда пользователь обращается с запросом к поисковой машине, он хочет найти то, что ему нужно, максимально быстро и просто. Получая результат, он оценивает работу системы, руководствуясь несколькими основными параметрами. Нашел ли он то, что искал? Если не нашел, то сколько раз ему пришлось переформулировать запрос, чтобы найти искомое? Насколько актуальную информацию он смог найти? Насколько быстро обрабатывала запрос поисковая машина? Насколько удобно были представлены результаты поиска? Был ли искомый результат первым или сотым? Как много ненужного мусора было найдено наравне с полезной информацией? Сможет ли он, вернувшись завтра и дав тот же запрос, получить те же результаты?

Для того, чтобы ответы на эти вопросы оставались удовлетворительными, разработчики поисковых машин постоянно совершенствуют алгоритмы и принципы поиска, добавляют новые функции, ускоряют работу системы. В этом реферате мы обратимся к механизму работы поисковой машины Рамблер, и на примере ее устройства продемонстрируем, как достигается повышение качества и скорости поиска в условиях постоянного роста объема информации в сети Интернет.

### Полнота

Полнота - это одна из основных характеристик поисковой системы, которая представляет собой отношение количества найденных по запросу документов к общему числу документов в Интернете, удовлетворяющих данному запросу. Например, если в сети Интернет имеется 100 страниц, содержащих словосочетание "Красная площадь", а по соответствующему запросу было найдено всего 70 из них, то полнота поиска будет 0,7. Чем полнее поиск, тем меньше вероятность, что пользователь не сможет найти нужный ему документ, при условии, что он вообще существует в Интернете.

Полнота поиска в большой мере зависит от работы системы сбора и обработки информации. В связи с постоянным ростом количества документов в сети, эта система в первую очередь должна быть масштабируемой. В Рамблере масштабируемость достигается за счет параллельного исполнения задачи произвольным количеством машин.

Сбором информации занимается робот-паук, который обходит страницы с заданными URL и скачивает их в базу данных, а затем архивирует и перекладывает в хранилище суточными порциями. Робот размещается на нескольких машинах, и каждая из них выполняет свое задание. Так, робот на одной машине может качать новые страницы, которые еще не были известны поисковой системе, а на другой - страницы, которые ранее уже были скачаны не менее месяца, но и не более года назад. Хранилище у всех машин едино. При необходимости работу можно распределить другим способом, например, разбив список URL на 10 частей и раздав их 10 машинам. Параллельная работа программы позволяет легко выдерживать дополнительную нагрузку: при увеличении количества страниц, которые нужно обойти роботу, достаточно просто распределить задачу на большее число машин.

В хранилище информация в сжатом виде собирается и разбивается на куски по 50 Мб. Эти части постепенно распределяются между 70 машинами, на которых запущена программа-индексатор. Как только индексатор на одной из машин заканчивает обработку очередной части страниц, он обращается за следующей порцией. В результате на первом этапе формируется много маленьких индексных баз, каждая из которых содержит информацию о некоторой части Интернета. Таким образом, вся интеллектуальная обработка данных осуществляется параллельно, поэтому ускорение процесса индексации достигается простым добавлением машин в систему.

После того, как все части информации обработаны, начинается объединение (слияние) результатов. Благодаря тому, что частичные индексные базы и основная база, к которой обращается поисковая машина, имеют одинаковый формат, процедура слияния является простой и быстрой операцией, не требующей никаких дополнительных модификаций частичных индексов. Основная база участвует в анализе как одна из частей нового индекса. Так, если объединяются 70 новых частей, то в анализе участвует 71 фрагмент (70 новых + основная база предыдущей редакции). Кроме того, единый формат позволяет проводить тестирование частичных баз еще до объединения их с основной, и обнаруживать ошибки на более раннем этапе.

Специальная программа ("сливатор") составляет таблицы перенумерации документов базы. Содержимое всех частей объединяется. Среди страниц с одинаковыми адресами выбирается наиболее свежая версия; если при скачивании URL последней информацией была ошибка 404 (запрашиваемая страница не существует), она временно удаляется из индексной базы. Параллельно осуществляется склейка дублей: страницы, которые имеют одинаковое содержимое, но различные URL, объединяются в один документ.

Сборка единой базы из частичных индексных баз представляет собой простой и быстрый процесс. Сопоставление страниц не требует никакой интеллектуальной обработки и происходит со скоростью чтения данных с диска. Если информации, которая генерируется на машинах-индексаторах, получается слишком много, то процедура "сливания" частей проходит в несколько этапов. В начале частичные индексы объединяются в несколько промежуточных баз, а затем промежуточные базы и основная база предыдущей редакции пересекаются. Таких этапов может быть сколько угодно. Промежуточные базы могут сливаться в другие промежуточные базы, а уже потом объединяться окончательно. Поэтапная работа незначительно замедляет формирование единого индекса и не отражается на качестве результатов.

### Точность

Точность - еще одна основная характеристика поисковой машины, которая определяется как степень соответствия найденных документов запросу пользователя. Например, если по запросу "Красная площадь" находится 150 документов, в 70 из них содержится словосочетание "Красная площадь", а в остальных просто присутствуют эти слова ("**красная** баба кричала на всю **площадь**"), то точность поиска считается равной 70/150 (~0,5). Чем точнее поиск, тем быстрее пользователь находит нужные ему документы, тем меньше "мусора" среди них встречается, тем реже найденные документы не соответствуют запросу.

Повышение точности в поисковой машине Рамблер достигается за счет использования различных технологий на всех этапах обработки и поиска информации. Одним из наиболее интересных процессов является распознавание грамматических омонимов. Омонимы - это слова, которые имеют одинаковое написание, но различный смысл. Различают лексические и грамматические омонимы. Лексические омонимы относятся к одной части речи, как, например, существительное "бор": хвойный лес, стальное сверло и химический элемент. Грамматические омонимы относятся к разным частям речи, поэтому по написанию у них обычно совпадают только отдельные формы. Примерами грамматических омонимов могут служить слова "печь" - существительное русская "печь" и глагол "печь" пирожки; "рядовой" - прилагательное "рядовой" сотрудник и существительное "рядовой" Иванов.

Омонимы не только увеличивают размер индексной базы (так как для каждого такого слова приходится хранить все его возможные значения), но и отрицательно сказываются на точности поиска. Если пользователь ищет слово "данные", ему неинтересно получить в найденном все документы, которые содержат слово "дать". Для того, чтобы результаты поиска были точнее, модуль синтаксического анализа проводит разбор окружения слов-омонимов с целью установления их наиболее вероятных значений. Например, если рядом со словом "печь" стоит существительное ("пирожки", "картошка"), то с высокой вероятностью "печь" в данном контексте является глаголом. На сегодняшний день анализатор способен распознавать значения только грамматических омонимов.

Синтаксический анализ позволяет также с определенной вероятностью распознавать некоторые имена собственные. Например, если в тексте несколько слов подряд написано с большой буквы, они чаще всего представляют собой имя собственное (Петр Петрович, Московский Государственный Университет). Данные о таких конструкциях учитываются при индексации и обработке запроса.

Еще один способ повышения точности поиска - это выделение устойчивых обозначений и поиск их как отдельных лексических единиц. На сегодняшний день в Рамблере реализована система распознавания таких конструкций, например C++, б/у, п/п-к. Если по запросу С++ поднимать все тексты, в которых присутствуют латинская буква С, а также знак +, то получится огромное количество документов, далеко не все из которых соответствуют запросу; кроме того, это большая работа, значительно увеличивающая время поиска.

Огромную роль в повышении точности поиска играет ранжирование. Пользователь очень редко просматривает больше трех страниц с результатами поиска. Поэтому субъективно он оценивает точность по "верхним" документам. Даже если нужный документ найден поисковой машиной, но расположен на двухсотой позиции, скорее всего, он никогда не будет найден пользователем.

По умолчанию в Рамблере результаты ранжируются по степени соответствия (релевантности) запросу и группируются по сайтам. При ранжировании оцениваются различные характеристики текстов, такие как:

* Количество вхождений слов (словосочетаний) в документ - чем больше раз словосочетание "Красная площадь" присутствует в тексте, тем выше вероятность, что в нем действительно говорится о Красной площади;
* Расположение слов запроса в документе - если словосочетание "Красная площадь" присутствует в заголовках или названии документа, то документ с большей вероятностью посвящен Красной площади;
* Формы слов запроса - преимущество отдается вхождениям, в которых слова имеют тот же падеж, число, склонение и т.д., что и в запросе пользователя ("Красная площадь", а не "Красной площадью"). Помимо точного совпадения, выделяются две группы форм слов - близкие и далекие. Близкими считаются изменения по падежам, склонениям, спряжениям, числам и родам. Далекими формами являются причастия, деепричастия и т.п. При ранжировании преимущество отдается близким формам слов запроса.
* Расстояние между словами запроса - если запрос состоит из нескольких слов, то в найденных документах оценивается, насколько близко друг от друга расположены эти слова. Преимущество отдается документам, в которых слова запроса находятся ближе друг к другу, потому что в этом случае они с большей вероятностью связаны между собой. Например, если слово "Красная" расположено в тексте на 5 позиции, а слово "площадь" - на 650, то скорее всего в документе речь идет не о Красной площади.
* Относительная частота (отношение количества вхождений слов запроса в документ к общему количеству слов в документе) - если словосочетание встречается 10 раз в документе из 100 слов, то он скорее соответствует запросу, чем если оно встречается те же 10 раз в документе из 20 тысяч слов;
* Популярность - поисковая машина автоматически вычисляет коэффициент популярности каждой страницы Интернет на основе данных счетчика Top100 и анализа гипертекстовых ссылок между страницами. Преимущество отдается более популярным ресурсам.
* Ссылочный вес документа - при ранжировании учитывается ссылочный вес страницы, рассчитанный на основании учета гиперссылок, содержащих слова запроса. Так, если на некоторый документ словами "Красная площадь" ссылается большое количество страниц с высокими поэффициентами популярности, то ему отдается приоритет по запросу Красная площадь.

Помимо автоматических способов увеличения точности поиска, существуют различные средства, с помощью которых пользователь сам может уточнить поиск по отдельным запросам. В первую очередь к ним относится специальный язык поискового запроса, используя который можно ограничивать количество найденных документов. Например, запрос или его часть, взятые в кавычки, обрабатываются буквально, с учетом всех стоп-слов, форм, порядка, знаков препинания. Это повышает точность поиска, но уменьшает его полноту: если часть, заключенная в кавычки, неточна, нужный документ найден не будет.

Использование логического оператора OR (ИЛИ) позволяет расширить сферу поиска и увеличить его полноту, в то время как оператор NOT (И-НЕ), наоборот, повышает точность поиска за счет нахождения документов, которые содержат одни слова запроса и не содержат другие. Для повышения точности можно также задавать расстояние между словами. Если в искомом словосочетании порядок слов обычно сохраняется (например, Красная площадь), то в запросе для повышения точности имеет смысл ограничить расстояние, указав его в скобках через запятую: (2, Красная площадь). Это позволит отсеять документы, в которых слова красная и площадь не расположены рядом, а разбросаны по тексту.

Увеличить точность можно с помощью использования поиска в найденном. Уточняющий поиск, проводится уже не по всей индексной базе, а только по результатам предыдущего поиска. Таким образом, круг найденных документов сужается. Например, если дать запрос Красная площадь, а затем, провести поиск в найденном по запросу Москва, то результат будет содержать только те документы, в которых говорится о Красной площади города Москвы.

### Актуальность

Актуальность - не менее важная характеристика поиска, которая определяется временем, проходящим с момента публикации документов в сети Интернет, до занесения их в индексную базу. Например, на следующий день после теракта в Тушино огромное количество пользователей обратились к поисковой машине Рамблер с соответствующими запросами. Объективно с момента публикации новостной информации на эту тему прошло меньше суток. Однако основные документы уже были заиндексированы и доступны для поиска, благодаря существованию "быстрой базы", которая обновляется два раза в день, а при необходимости может обновляться быстрее.

На сегодняшний день индексная база поисковой системы Рамблер состоит из 8 частей, каждая из которых живет своей независимой жизнью. Весь Интернет условно разделен на 7 секторов и называется своим цветом: красный, оранжевый, желтый, зеленый, голубой, синий, фиолетовый. Сайт компании Рамблер относится к голубому сектору. Информация о web-ресурсах каждого сектора хранится в соответствующей части индексной базы. Восьмая часть - "быстрая база" - включает в себя страницы, на которых размещен счетчик Тор 100 и которые еще не успели попасть в основную индексную базу.

Все части индексной базы собираются и обновляются по отдельности. Так, сегодня происходит переиндексация и обновление красного сектора, завтра - оранжевого и желтого, послезавтра - зеленого и т.д. Благодаря такому ступенчатому алгоритму в поисковой машине регулярно появляется свежая информация. Полный цикл обновления занимает около недели. При этом сбор информации происходит параллельно, а непосредственно на изготовление индекса документов одного сектора уходит всего несколько часов. Поэтому существует принципиальная возможность обновлять индексную базу быстрее.

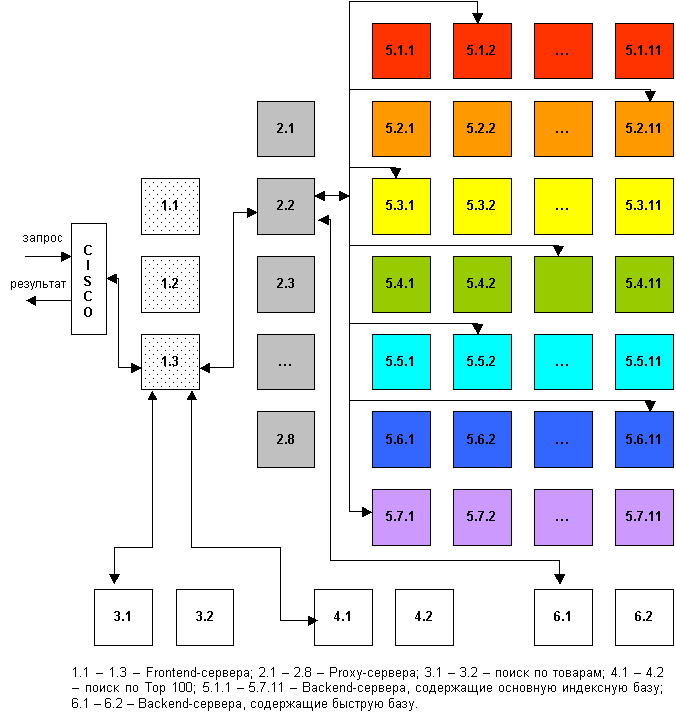
Разделение Интернета на 7 секторов условно. При необходимости он может быть разбит на 10, 20 или 40 секторов, каждый из которых будет обрабатываться автономно. В такой системе заложена возможность значительного увеличения нагрузки. С ростом объема информации в сети Интернет растет и индексная база поисковой машины. Постепенно переиндексация и сборка базы начинает занимать все больше времени, а процесс обновления индекса становится более громоздким. Поступление новых данных затягивается, информация начинает терять свою актуальность. Возможность "передела" Интернета на большее число секторов позволяет удерживать размер каждой части базы в оптимальном диапазоне, контролировать время ее сборки и обновления.

"Быстрая база" отличается от остальных частей индекса меньшим объемом и очень оперативным обновлением: время ее построения занимает около двух часов. В базе содержится информация о страницах, на которых был установлен счетчик Тор 100. Участниками рейтинга Тор 100 являются новостные порталы, сайты крупных компаний, Интернет-магазины, форумы, - все наиболее популярные ресурсы в сети. Каждый раз при установке счетчика на новую страницу сайта, зарегистрированного в Тор 100, информация передается в поисковую систему. Страница ищется во всех цветах основной базы и, если она еще не известна поисковой системе, отправляется в очередь на обработку. Перед обработкой страницы дополнительно фильтруются, из них отбираются самые посещаемые. Таким образом, "сливки" с Интернета собираются два раза в день.

"Быстрая база" представляет собой разумное решение проблемы актуальности данных в поиске. Информационное агентство может выложить новость через десять минут после ее появления, потому что тратит время только на верстку страницы. Поисковая машина должна сначала заиндексировать текст, а на это требуется гораздо больше времени. "Быстрая база" охватывает все ресурсы Интернет, зарегистрированные в Тор 100, на которых был размещен счетчик, и которые еще не успели попасть в основную базу. При этом индексируются как страницы с новостями, так и другие свежие документы, появившиеся в Тор 100. В результате через сутки после теракта в поиске Рамблера была доступна не только основная информация, опубликованная на сайтах новостных агентств, которую можно найти и прочитать в разделах новостей, но и комментарии, высказывания очевидцев, обсуждения на форумах, все, что было к этому времени опубликовано на наиболее посещаемых страницах Интернета.

### Скорость поиска

Скорость поиска тесно связана с его устойчивостью к нагрузкам. На сегодняшний день в рабочие часы к поисковой машине Рамблер приходит около 60 запросов в секунду. Такая загруженность требует сокращения времени обработки отдельного запроса. Здесь интересы пользователя и поисковой системы совпадают: посетитель хочет получить результаты как можно быстрее, а поисковая машина должна отрабатывать запрос максимально оперативно, чтобы не тормозить вычисление следующих. Схематично обработка поискового запроса изображена на рисунке 1.



Запрос поступает в поисковую систему через маршрутизатор Cisco 6000 series. Cisco передает его наименее загруженной машине первого уровня - frontend (1.1 - 1.3, на рис. машине 1.3). Frontend, в свою очередь, отправляет запрос дальше, на один из восьми proxy-серверов, также выбирая наиболее свободный сервер (2.1 - 2.8, на рис. машине 2.2). Одновременно frontend отправляет запрос на машины, осуществляющие поиск по товарам (3.1 - 3.2, на рис. машине 3.1) и по базе Тор 100 (4.1 - 4.2, на рис. машине 4.1). На proxy проводится поиск по ссылочному индексу, и его результаты вместе с поисковым запросом передаются на машины, которые содержат основную индексную базу, - backends (5.1.х - 5.7.х, на рис. машинам 5.1.2, 5.2.11, 5.3.1 и т.д.) Та же информация отправляется на машины с "быстрой базой" (6.1 - 6.2, на рис. 6.1).

На текущий момент в поиск включено 77 backend'ов. Они сгруппированы по 11 машин, и каждая группа содержит копию одной из частей поискового индекса. Таким образом, информация о сайтах, условно входящих в красный сектор Интернета, находится на backend'ах первой группы (5.1.1 - 5.1.11 на рис), оранжевый сектор - на backend'ах второй группы (5.2.1 - 5.2.11) и т.д. Proxy-сервер выбирает наименее загруженный backend в каждой группе машин и отправляет на него поисковый запрос с результатами ссылочного поиска. На backend'ах осуществляется поиск по частям индексной базы и ранжирование с учетом результатов поиска по ссылочному индексу. При ранжировании для всех найденных документов высчитываются веса по конкретному запросу.

После того, как запрос обработан на backend'ах, информация о результатах и ранжировании отдается обратно на proxy-сервер. Туда же поступают отсортированные результаты с машин "быстрой базы". Proxy интегрирует данные, полученные с восьми машин: клеит дубли, объединяет зеркала сайтов, переранжирует документы в общий список по весам, рассчитанным на backend'ах. Так, первым в списке найденного может быть документ с машины 5.3.1, вторым и третьим - с 6.1, четвертым - с 5.5.2 и т.д. На proxy-сервере также реализуется построение цитат к документам и подсветка слов запроса в тексте. Полученные результаты отдаются на frontend.

Помимо информации с proxy-сервера, frontend получает результаты из поиска по товарам и из базы Тор 100, отсортированные, с цитатами и подсветкой слов запроса. Frontend осуществляет окончательное объединение результатов, генерирует html со списком найденного, вставляет баннеры и перевязки (ссылки на различные разделы Рамблера) и отдает html Cisco, который маршрутизирует информацию пользователю.

Каждый из этапов обработки запроса многократно продублирован и защищен системой балансировки нагрузки. Благодаря дублированию информации поисковая система Рамблер является устойчивой к сбоям на отдельных участках, авариям, отказам оборудования. Если одна их машин перестала функционировать, нагрузка перераспределяется на другие машины, и выпадения документов из поиска не происходит. Масштабируемость достигается простым добавлением в систему машин соответствующего уровня. До недавнего времени в Рамблере работало 45 backend'а. В связи с тем, что осенью нагрузка на поисковые системы обычно возрастает, число backend'ов было увеличено до 77, что позволило значительно ускорить вычисление запросов.

Еще один способ повышения скорости поиска - "кэширование", сохранение информации о запросах и результатах поиска в буфере. Многие люди дают одни и те же поисковые запросы. Вычислять их каждый раз заново было бы неразумной тратой времени. Поэтому если запрос уже обрабатывался в течение некоторого интервала времени, результаты поиска отдаются пользователю из "кэша".

Лингвистический анализ текста документов и запроса также позволяет ускорить обработку информации. Например, определение значения омонимов уменьшает количество нерелевантных запросу документов, которые нужно ранжировать и цитировать. Выделение устойчивых обозначений (С++, б/у) на этапах индексации и обработки запроса приводит одновременно к повышению точности и сокращению временных затрат на обработку каждого отдельного элемента обозначения (раньше запрос С++ обрабатывался как отдельно латинское С, отдельно плюс и еще один плюс. Запрос вычислялся долго, а среди результатов поиска было много нерелевантных документов, например, страницы, содержащие математические формулы и т.п.) С этой же целью используются словари стоп-слов. Стоп-слова - это наиболее частотные слова языка, которые встречаются практически в любом тексте и являются малоинформативными. В основном, это служебные слова - предлоги, частицы, артикли. Если нет специальных указаний, поисковая машина игнорирует стоп-слова, встречающиеся в запросе, чтобы не тратить время на обработку дополнительной информации, снижающей качество поиска.

### Наглядность

Наглядность представления результатов является необходимым компонентом удобного поиска. На плохой витрине легко не заметить хороший товар. По большинству запросов поисковая машина находит сотни, а то и тысячи документов. В следствие нечеткости запросов или неточности поиска, даже первые страницы не всегда содержат только нужную информацию. Это означает, что пользователю часто приходится проводить свой собственный поиск внутри списка найденного. Различные элементы ответной страницы помогают ориентироваться в результатах поиска.

Группировка по сайтам предназначена для того, чтобы на странице можно было вывести как можно больше Интернет-ресурсов, релевантных запросу пользователя. Это бывает важным, когда необходимо получить информацию из различных источников. Если более информативной для посетителя является дата обновления или релевантность отдельных документов, в ответной странице Рамблера существует возможность сортировки по этим параметрам.

В некоторых случаях полезным бывает знание имени сайта. Если пользователя интересует конкретный Интернет-ресурс, имя может дать ему гораздо больше информации, чем заголовок страницы или цитата. Если запросу соответствует больше одной страницы с сайта, то в качестве результата поиска предъявляется наиболее релевантная из них, а ниже располагается частичный список остальных документов. Это увеличивает количество потенциально полезной информации на ответной странице и часто позволяет уточнить поиск без дополнительного запроса.

Цитата помогает определить, насколько полезную информацию содержит найденный документ. Очень часто посетителю не требуется переходить по ссылке, чтобы обнаружить, что текст не соответствует его интересам и потребностям. Иногда ответ на вопрос пользователя содержится непосредственно в цитате документа. Это экономит время и повышает эффективность работы поисковой системы.

Восстановить текст - иногда единственный способ получить доступ к содержимому найденного документа. Ресурс бывает недоступен по разным причинам. Документ может быть удален, перенесен, изменен, но его текстовое содержание некоторое время сохраняется в индексной базе. Кроме того, внутри самого документа часто отсутствует навигация, позволяющая быстро найти фрагмент, релевантный запросу. В восстановленном тексте все слова запроса подсвечиваются.

Ассоциации представляют собой список запросов, которые часто подаются пользователями в течении одной поисковой сессии. Алгоритм построения ассоциаций устроен так, что они почти всегда связаны между собой по смыслу. В некоторых случаях ассоциации позволяют повысить качество поиска за счет уточнения запроса (запрос "отдых в Польше" - ассоциации "отдых в Польше с детьми", "семейный отдых", "пансионаты в Польше"), исправления распространенных ошибок (запрос "gjujlf" - ассоциация "погода"), возможности сориентироваться в незнакомой тематике (запрос "антибиотик" - ассоциации "сумамед", "цифран", "бисептол" и т.д.)

### Заключение

Заключение пишется в конце и предполагает конечность. Но рост информации бесконечен, а потому нет предела совершенствованию поисковых машин. Важнейшей задачей разработчиков является улучшение качества поиска, движение в сторону большей эффективности и удобства в использовании системы. С этой целью постоянно меняются поисковые алгоритмы, создаются дополнительные сервисы, дорабатывается дизайн.

Однако для того, чтобы выжить в мире динамичного Интернета, при разработке необходимо закладывать большой запас устойчивости, постоянно заглядывать в завтрашний день и примерять будущую нагрузку на сегодняшний поиск. Все, что сегодня программируется в Рамблере, рассчитано "на вырост". Такой подход позволяет заниматься не только постоянной борьбой и приспособлением поисковой машины к растущим объемам информации, но и реализовывать что-то новое, действительно важное и нужное для повышения эффективности поиска в сети Интернет.

При подготовке реферата использовались официальные материалы компании «Rambler».