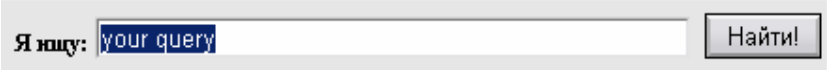
Использование Avalanche Personal в качестве персональной метапоисковой системы

Возможности персональной поисковой системы Avalanche Personal не ограничиваются только сбором сообщений с заданных источников. Существует уникальная возможность использовать продукт для поиска сообщений, используя информацию, собранную действующими поисковыми системами в сети Интернет. При этом пользователю предоставляется практически неограниченный выбор всевозможных поисковых систем: начиная с самых известных (www.yandex.ru, www.google.com, www.rambler.ru, …) и заканчивая поисковыми машинами отдельных сайтов интересующей Вас тематики. Таким образом, Вы можете использовать специализированные поисковые системы Интернета (www.sourceforge.net, www.ixbt.com, www.price.ru, news.yandex.ru), если Вас интересует информация, получаемая с их помощью. Фактически, если Вы увидели на странице в Интернете окошко «поиск», Вы можете настроить систему для получения сообщений с этого поискового движка, это справедливо и для поиска по форумам.

Модель поисковой системы

Модель использования поисковых систем в качестве источников сообщений следующая:

1. Когда Вы вводите запрос в поисковую систему Интернета и нажимаете кнопку «Найти»:



после получения результатов поиска Вы в адресной строке браузера видите примерно следующее:

http://search.rambler.ru/srch?words=your+query&where=1

Эта строка делится на три области:

1) Строка до запроса: **http://search.rambler.ru/srch?words=**
2) Закодированная строка запроса: **your+query**
3) Строка после запроса: **&where=1**

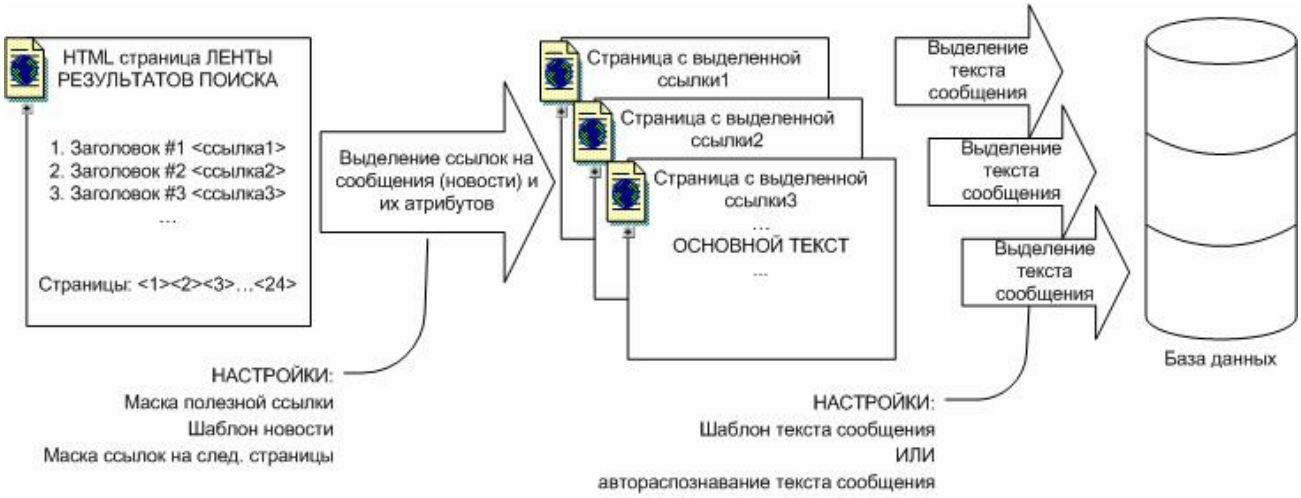
При добавлении нового поисковика в ваши настройки Avalanche, необходимо указать строку до запроса (1) и строку после запроса (3). Кодирование самой строки запроса (2) достаточно стандартизовано, поэтому Avalanche выполняет ее автоматически.

Иногда в поисковике существуют дополнительные опции, например поиск по дате, или количество результатов на страницу, при этом URL запроса изменяется (например, добавляется **&numdoc=20**). Вы можете использовать все эти опции, изменяя строку до запроса (1) и строку после запроса (3), которые Вы увидите в браузере при настройке системы.

2. Вторым этапом настройки для каждого поисковика, определенного согласно п.1, необходимо задать набор запросов (в терминологии программы «ссылок»). Для каждого запроса необходимо задать поисковую фразу. Если используются логические выражения, их нужно задавать в формате конкретного поисковика, так как система не может знать язык запросов для каждого поискового движка.

   При добавлении поисковика на этапе 1 вы можете установить опцию «сформировать запросы в соответствии с описанием рубрик». Это приведет к тому, что автоматически добавится набор запросов, содержащих булевы фильтры рубрик. Но при этом будьте внимательны: не многие поисковики поддерживают всю функциональность булева фильтра Avalanche, который Вы используете при настройке рубрик. Скорее всего, придется упростить запрос или изменить синтаксис булевых операций (AND, OR, NOT).

3. Третий этап заключается в тонкой настройке каждой ссылки, добавленной в поисковик. Для этого опишем общую схему работы паука для указанной ссылки.



Сбор сообщений состоит из следующих этапов:

1) Из Интернета достается страница – лента результатов поиска, URL которой получается из конкатенации (сложения) «строки до запроса», «закодированный Ваш запрос» и «строки после запроса» (см. п. 1)
2) Полученная страница анализируется и в ней выявляются ссылки, которые нашел поисковик. Вместе со ссылками могут выделяться также и другие атрибуты: заголовок (обычно текст, являющийся ссылкой, или рядом со ссылкой), дата (если есть), время (если есть), подзаголовок. Анализ основывается либо на внутренних эвристических правилах, который в некоторых случаях позволяют интеллектуально распознавать

ссылки и их атрибуты, либо на подсказах пользователя в виде маски ссылки, шаблона[1].

3) Паук проверяет каждую выделенную ссылку на предмет того, проходил ли он ее раньше и составляет список «новых» ссылок[2].

4) Паук обходит все новые ссылки, выделяя смысловой кусок текста, который становится текстом сообщения. При этом также используются интеллектуальные алгоритмы, либо пользователем задается шаблон текста сообщения. В этом случае, в отличие от страницы-ленты, алгоритм работает в 70-80% случаев. К тому же, если поиск проводится по разноформатным источникам, шаблон задавать бессмысленно. Если формат результатов поиска определен жестко (например, поиск проводится среди новостей одного сайта), то целесообразно задать шаблон текста сообщений для гарантированного результата и увеличения скорости работы системы.

5) Если задана маска ссылок на следующие страницы, то на исходной странице-ленте находится еще не обследованная ссылка, удовлетворяющая маске. В случае если количество уже обработанных лент не превышает значения поля «глубина» в настройках, по найденной ссылке получается «продолжение контента» и обрабатывается аналогично, начиная с п. 2). Если маска на следующие страницы отсутствует, или глубина установлена в «1», то паук прекращает сбор сообщений с этой ссылки (по этому запросу).

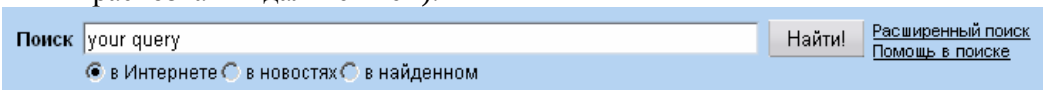
[1] На практике, для качественного сбора сообщений без потерь рекомендуется задавать шаблон при настройке ссылки.

[2] Вся история пройденных ссылок храниться в файлах вида http__* в папке базы данных. Если Вы хотите, чтобы паук снова обошел уже пройденные ссылки с какого-то сайта, Вам надо удалить соответствующий файл. Удаления собранных сообщений для этого недостаточно.

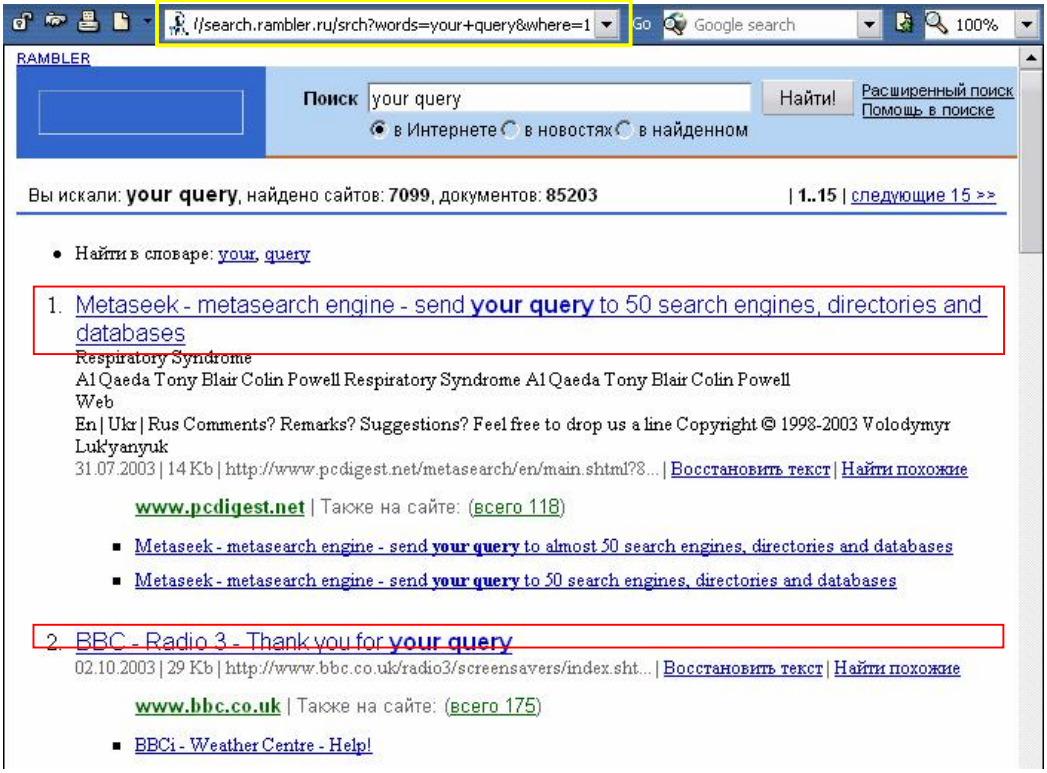
Пример настройки системы

В качестве примера, продемонстрируем процесс настройки Avalanche Standard на поиск через поисковую систему www.rambler.ru.

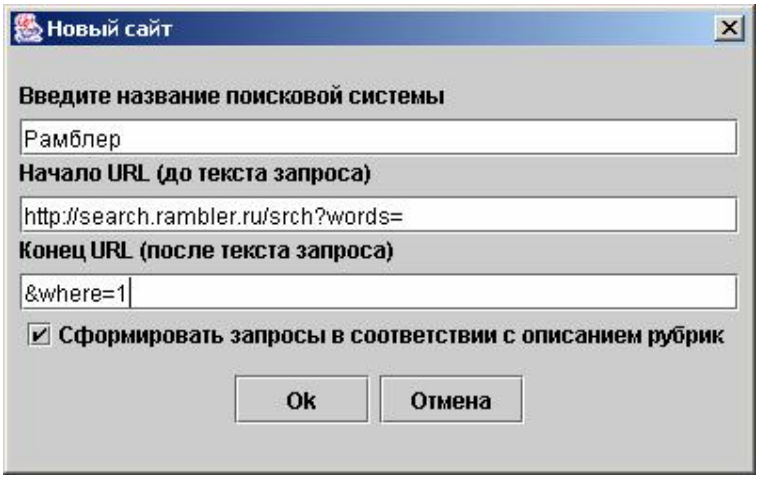
1. Откроем страницу www.rambler.ru в браузере и введем тестовую поисковую фразу на английском языке (для того, чтобы легко ее распознать в дальнейшем):



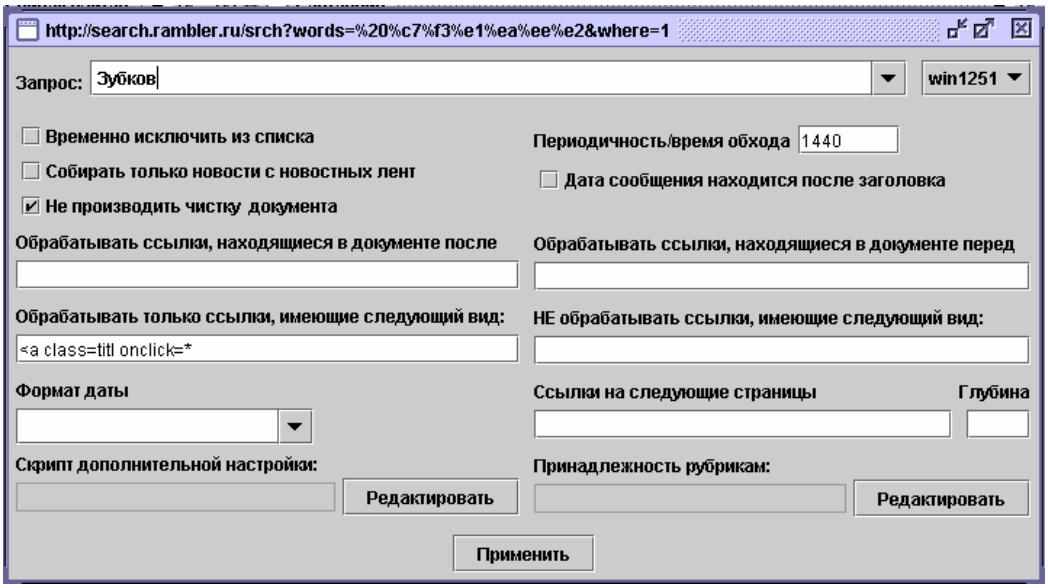
2. После нажатия кнопки «Найти», отобразится страница с результатами поиска:



В поле адреса браузера (желтая рамка на рисунке) находится следующая строка:
http://search.rambler.ru/srch?words=your+query&where=1
Выделив наш запрос, определяем, что
- строка до запроса: http://search.rambler.ru/srch?words=
- строка после запроса: &where=1
В программе «Паук» щелкнув по корню списка сайтов и выбрав «добавить поисковик», вводим:
- название поисковой системы: Рамблер
- строку до запроса
- строку после запроса



После этого, нажав Ok, получим новый поисковик с уже определенными ссылками (запросами) по количеству рубрик. Открываем свойства ссылки (уже сформированной, или добавленной вручную для этого поисковика)



(см. «Справочную информацию»)

Можно выделить три основных этапа в процессе настройки сбора сообщений, которые также можно назвать «уровнями автоматизации сбора»:
1 уровень («автомат») – Вы снимаете «собирать с новостных лент»[3], устанавливаете при необходимости «дата после заголовка», указываете формат даты. Этот уровень не требует знания HTML, достаточно только взглянуть на страницу, чтобы определить эти параметры, которые будут являться подсказками алгоритму распознавания ленты.
2 уровень («полуавтомат»)– Кроме перечисленных выше параметров, вы также указываете маски обрабатываемых ссылок («обрабатывать только

[3] В редких случаях на первом уровне сбор начинает работать при включенной «собирать с новостных лент»

ссылки имеющие следующий вид») и, возможно, фрагменты HTML, обрезающие контент ленты (см. «Справочную информацию»). Этот уровень уже требует анализа HTML-кода.

3 уровень («ручной»)– когда Вы задаете шаблон ленты в «скрипте дополнительной настройки». Этот код требует основательного анализа HTML-кода страницы, на предмет поиска подходящего шаблона, который бы охватывал все ссылки и при этом не захватывал лишние.

Чаще всего достаточно второго уровня, но третий уровень повышает качество сбора (например появляется возможность указать где находится дата и время, или заголовок сообщения).

Независимо от уровня вы можете указывать маски ссылок на следующие страницы, глубину и шаблон текста новости.

Шаблон текста имеет смысл применять в следующих случаях:

1. Тексты всех сообщений имеют единый формат (то есть расположены на одном сайте и оформлены в одном стиле)
2. В автоматическом режиме иногда теряется конец или начало текста сообщения
3. В автоматическом режиме иногда забирается не тот текст со страницы
4. Сбор работает очень медленно (более полуминуты на сообщение)

Вернемся к исследуемой странице. На ней нас интересуют области, на которых находятся ссылки на результаты поиска. Для настройки системы нам необходимо увидеть HTML-код страницы (обычно меню view/source, вид/исходник в браузере). В исходнике нужно найти один участок HTML-кода, отвечающий за выделенный в красную рамку фрагмент страницы. В данном случае, фрагмент кода во второй рамке:

```
<li class=fsite><a class=titl onclick="R(this, 'r=31E0E669')"
href="http://www.bbc.co.uk/radio3/screensavers/index.shtml"
target=_blank>BBC
- Radio 3 - Thank you for <B>your</B> <B>query</B></a><br><font
size=-1>
```

В данном случае интересующая нас ссылка задается кодом:

```
<a class=titl onclick="R(this, 'r=31E0E669')"
href="http://www.bbc.co.uk/radio3/screensavers/index.shtml"
target=_blank>
```

Проверка по документу показывает, что строка class=titl (отвечающая за стиль) встречается в документе ровно 15 раз и определяет стиль интересующих нас ссылок. Таким образом, мы можем задать следующую маску ссылки в поле «обрабатывать только ссылки имеющие следующий вид»:

```
<a class=titl * href=* target=_blank>
```

Единственный метасимвол – «*» говорит о том, что нужно пропустить все символы, пока не встретим строку после * в шаблоне.

3. На одной странице рамблер выдает только 15 результатов. Для того, чтобы собрать результаты с более высоким номером, нам надо указать маску ссылок на следующие страницы и глубину (то есть количество анализируемых страниц). Аналогично находим код ссылки на вторую страницу:

```
<a href="/srch?words=your+query&amp;start=16">
```

и формируем маску:

```
<a href="/srch?words=*start=*
```

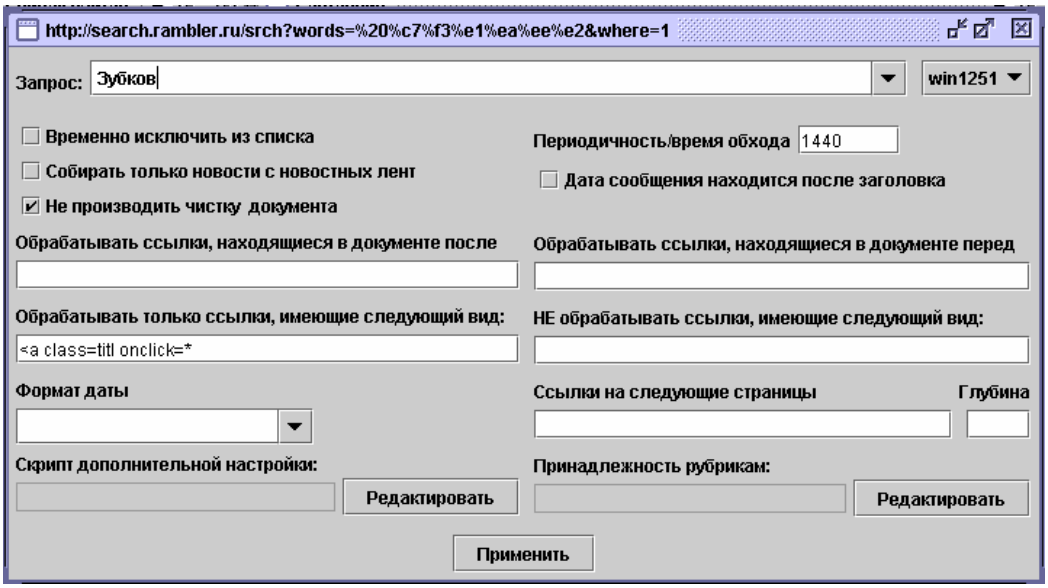
При формировании маски не забывайте проверять, что ей удовлетворяют только те ссылки, которые Вас интересуют.
Глубину можно поставить «2», чтобы собиралось 15*2 = 30 сообщений.

4. В данном примере мы настроили Рамблер по уровню 2 («полуавтомат»)[4]. Осталось только снять галочку «собирать только с новостных лент», так как мы не используем шаблон тонкой настройки $NewsShablon. И на всякий случай поставить галочку «Не производить чистку документа».

[4] Уровень 3 в данном случае позволил бы получить дату, которая находится на некотором расстоянии от заголовка результата поиска.

Справочная информация

Настройка ссылок



- В поле «запрос» можно изменить поисковую фразу, которая будет посылаться Рамблеру.
- Слева от поля запроса можно выбрать кодировку запроса. Чаще всего используется Windows-1251, но на некоторых поисковиках требуется посылать запрос в KOI-8. Определите это по кодировке странице результатов поисковика в браузере (например, изменив кодировку).
- Можно «временно исключить из списка» данную ссылку, чтобы она не участвовала в сборе сообщений
- «Периодичность/время обхода» задает либо интервал (в минутах), либо время в формате чч:мм, определяющее то, когда в следующий раз будет обрабатываться ссылка в режиме сбора «по таймеру»
- «Собирать новости только с новостных лент» - опция, переключающая алгоритмы распознавания лены новостей. Общая рекомендация – включать опцию, если используется шаблон ленты $NewsShablon, и выключать, если он не используется.
- Иногда «Дата сообщения находится после заголовка». Если это так, укажите эту опцию – возможно это позволит автоматически распознать ленту.
- «Не производить чистку документа» необходимо отмечать практически всегда, особенно если Вы указываете шаблоны.
- «Обрабатывать ссылки, находящиеся в документе после…» позволит отсечь ненужное начало HTML-кода страницы.
- «Обрабатывать ссылки, находящиеся в документе перед…» аналогично отсекает все после встреченного заданного фрагмента.

- «Обрабатывать только ссылки, имеющие следующий вид». Здесь укажите маску ссылки (например, <a class=someclass href=*). Звездочка обозначает произвольную цепочку. При анализе ленты новостей будут обрабатываться только ссылки, соответствующие указанной маске.
- «Не обрабатывать только ссылки, имеющие следующий вид». Аналогично, но удовлетворяющие маске ссылки наоборот не обрабатываются.
- «Формат даты» нужен для облегчения поиска даты и автоматического распознавания ленты пауком.
- «Ссылки на следующие страницы» определяемые задаваемой маской будут рассматриваться как ссылки на продолжение ленты. Формат маски такой же, как и раньше.
- «Глубина» определяет максимальное количество обрабатываемых дополнительно лент, ссылки на которые удовлетворяют указанной выше маске.
- Скрипт дополнительной настройки может содержать элементы $NewsShablon (задает шаблон новостной ленты) и $TextShablon (задает шаблон текста сообщений). Для того чтобы шаблон в встроенном редакторе сохранялся необходимо нажимать кнопку «сохранить» на панели инструментов редактора.
- Принадлежность рубрикам позволяет отметить те рубрики, в которые автоматически будут попадать все сообщения с настраиваемой ссылки.

Формат шаблона новостной ленты

$NewsShablon <значение> - позволяет полностью определять новость.

Пример:

$NewsShablon <table border=0 cellpadding=0 cellspacing=5>

<tr valign=top><td width=15>**~time~**</td>

<td width=1><img border=0 src="img/2.5/dot.gif" width=1></td>

<td width=615> **~title~** **~text~**</td>

</tr><tr valign="top"><td colspan="2"> </td><td align="right">

<table cellspacing=0 cellpadding=0 border=0 width="100%"><tr valign="top">

<td align="left" width="50%">**~something~**</td><td align="right" width="50%">**~something~**</td></tr></table></td></tr></table>

HTML-код, описывающий несколько новостей на одной странице, практически идентичен, поэтому в HTML- коде необходимо выявить повторяющиеся куски, скопировать его и заменить изменяющиеся участки на соответствующие параметры:

~something~ меняющийся текст который пропускается до слудующего за ним элемента в шаблоне.

~title~ текст, стоящий на этом месте, является заголовком сообщения.

~subtitle~ текст, стоящий на этом месте, является подзаголовком сообщения.

~date~ текст, стоящий на этом месте, является датой сообщения.

~time~ текст, стоящий на этом месте, является временем сообщения.

~url~ текст, стоящий на этом месте, является адресом страницы, на которой находится текст сообщения.

~text~ текст, стоящий на этом месте, является текстом сообщения.

~source~ источник сообщения.

~author~ автор сообщения.

Формат текстового шаблона

$TextShablon <значение> - позволяет полностью определять вид документа, содержащего текст сообщения. Параметры те же, что и в новостном шаблоне кроме **~url~** и **~subtitle~**(их нет)

$# - комментарии.

Внимание! При использовании скрипта тонкой настройки необходимо «не производить чистку документов»!