**Tuba search engine manual**

**Sawsan Dwiekat**

1. Introduction ………………………………………………………. 2
2. Install Tuba search engine …………………………………………2
3. Indexing options …………………………………………………...3
	1. full index
	2. depth index
	3. re-index
	4. indexer can leave domain
4. Keep pages , words , files from being indexed ……………………3
	1. robots.txt
	2. ignoring links
	3. canonical <link> tag
	4. ignoring part of the page
	5. ignored words
	6. ignored files
5. UTF8 multi language support ……………………………………..4
6. File convertors …………………………………………………….4
7. Search modes ……………………………………………………...5
8. Media search for images , video , audio ………………………….6
	1. media indexing
	2. search for media contents
9. RSS and atom feeds ……………………………………………….7
10. **Introduction :**

**Tuba** is a web based search engine so What is a search engine : Internet search engines are special sites on the Web that are designed to help people find information stored on other sites. There are differences in the ways various search engines work, but they all perform three basic tasks:

* They search the Internet -- or select pieces of the Internet -- based on important words.
* They keep an index of the words they find, and where they find them.
* They allow users to look for words or combinations of words found in that index.

**How do search engine work** ?

Before a search engine can tell you where a file or document is, it must be found. To find information on the hundreds of millions of Web pages that exist, a search engine employs special software robots, called **spiders**, to build lists of the words found on Web sites. When a spider is building its lists, the process is called **Web crawling**. (There are some disadvantages to calling part of the Internet the World Wide Web -- a large set of arachnid-centric names for tools is one of them.) In order to build and maintain a useful list of words, a search engine's spiders have to look at a lot of pages.

How does any spider start its travels over the Web? The usual starting points are lists of heavily used [servers](http://computer.howstuffworks.com/web-server.htm) and very popular pages. The spider will begin with a popular site, indexing the words on its pages and following every link found within the site. In this way, the spidering system quickly begins to travel, spreading out across the most widely used portions of the Web.

When the Google spider looked at an [HTML](http://computer.howstuffworks.com/web-page.htm) page, it took note of two things:

* The words within the page
* Where the words were found

**Meta tags** allow the owner of a page to specify key words and concepts under which the page will be indexed. This can be helpful, especially in cases in which the words on the page might have double or triple meanings -- the [meta tags](http://computer.howstuffworks.com/web-page6.htm) can guide the search engine in choosing which of the several possible meanings for these words is correct. There is, however, a danger in over-reliance on meta tags, because a careless or unscrupulous page owner might add meta tags that fit very popular topics but have nothing to do with the actual contents of the page. To protect against this, spiders will correlate meta tags with page content, rejecting the meta tags that don't match the words on the page.

1. **Installing Tuba search engine :**

First of all , you must create database at your server , name it whatever you want , but make sure that your database has the UTF8\_bin collation .

1. Copy all the files in CD into your server
2. Go to settings folder at your server , open the file database.php and change the database name to the name of the database you have already created.
3. Open your browser and go to your server address something like this <http://localhost/> go to Admin folder and open tables.php into your browser with url something like <http://localhost/public/tuba/admin/tables.php> this process will install all the tables needed into your database , make sure to fill [info] table in your database with information needed (name , password)
4. Now your database is ready to start indexing , open your browser , go to your server , tuba , then to admin folder and open admin\_form.php , type your log in information , enjoy indexing whatever sites you need ☺

1. **Indexing options :**

As part of the Admin Site settings you may select the following options:

**Full** **Index**: Indexing continues until there are no further (permitted) links to follow.

**Index Depth**: Indexes to a given depth, where depth means how many "clicks" away the page can be from the starting page. Depth 0 means that only the starting page is indexed, depth 1 indexes the starting page and all the pages linked from it etc.

**Re-index**: By selecting this mod, indexing is forced even if the page already has been indexed. Re-index only detects **changes** of the pages to be re-indexed. Modifications in Admin settings are not recognized.

**can leave domain:**

By default, Tuba never leaves a given domain, so that links from domain.com pointing todomain2.com are not followed. By checking this option Tuba can leave the domain , and this option can be used to index specific web site without leaving its domain , with full index option checked , tuba can work as specific search engine for some web site and not general .

1. **Keeping pages , words and files from being indexed :**

 Tuba respects the websites rules in indexing in which search engines allowed or prevented from following some links , these rules are :

* robots.txt :

The most common way to prevent pages from being indexed is using the robots.txt standard, by either putting a robots.txt file into the root directory of the server, or adding the necessary meta tags into the page headers.

* Ignoring links:

Tuba respect rel="nofollow" attribute in <a href . . . > tags, so for example the link foo.html in <a href="foo.html" rel="nofollow"> will be ignored. Also if the nofollow flag is set in the header of a site, this link will not been followed.

* Canonical <link> tag : As defined by Google, Microsoft and Yahoo! in February 2009, also Tuba will follow the instruction of a rel="canonical" link. You may simply add this <link> tag to specify your preferred page version:

<link rel="canonical" href="http://www.example.com/product.php?item=swedish-fish" /> inside the <head> section of all the duplicate content URLs:

<http://www.example.com/product.php?item=swedish-fish&category=gummy-candy> <http://www.example.com/product.php?item=swedish-fish&trackingid=1234&sessionid=5678> and Tuba will understand that the duplicates all refer to the canonical URL: http://www.example.com/product.php?item=swedish-fish.

The duplicate pages will be ignored and not indexed. Tuba takes the rel="canonical" as a directive, not a hint. The canonical link may also be a relative path, but is not allowed to refer to a different domain. Unfortunately the creation of canonical link tags needs to be done manually. So special care has to be taken that other directives like robots.txt or rel="nofollow" will not prevent the crawling of the canonical origin.

* Ignoring parts of a page : Tuba includes an option to exclude parts of pages from being indexed. This can for example be used to prevent search result flooding when certain keywords appear on certain part in most pages (like a header, footer or a menu). Any part of a page between <!-- --> and <!--/ --> tags is not indexed, however links in it are followed.
* Ignored words : Tuba offers the capability to prepare language specific common files. Common words that are not to be indexed can be placed into individual files. The names of this files must start with 'common\_' and end with the suffix '.txt', like "common\_eng.txt .
* Ignored files: The list of file types that are not checked for indexing are places in .../include/common/ext.txt. This file holds all file suffixes for those type of files that are to be ignored during index / re-index procedure. The 'ext.txt' file is independent of the media files to be indexed. All file types not to be followed for text indexing must be placed in 'ext.txt'. To be seen as a blacklist for file suffixes. While image.txt audio.txt video.txt are lists that include suffixed for files to be indexed according to the type of media.
1. **UTF8 and multi language support :**

Tuba search is multi language , so you can search either by Arabic or English language , this can be achieved by indexing both English and Arabic sites , to do so and store them in the same tables in database Tuba uses Unicode assistance in which 63 charsets are supported and will be converted into UTF-8 Unicode , all these types of charset can be found at …… /convertors/charsets

**To convert any charset into UTF8 , Tuba proceeds the following :**

1. Detect charset of site, page or file that's content has to be translated.

This information is normally presented as part of the HTML header. If not available, or for files without header like .doc, .rtf, .pdf, .xls and .ptt files, the 'Preferred charset' will be used to translate the file into Unicode.

 In other words: you can't convert DOCs, PDFs etc. that are coded in 'foreign' charset. Only those with your personal charset will be converted correctly.

2. By means of the PHP function 'iconv()' content and keywords will be converted into UTF-8. This step is successful, if the required charset (for the content to be translated) is part of your local PHP installation. In order to find out which charset are available in your installation, notice the files in server folder: .... .../apache/bin/iconv/ Depending of the installation you will find about 200 charset files that iconv() is able to translate into UTF-8

3. If the PHP function fails, finally the class 'ConvertCharset' is invoked. This class, enables translation for a lot of charset. But it takes more time than the compiled PHP function 'iconv()'. As result of this translation, you are also enabled to search for words that contain non-Latin characters.

 **6. File convertors :**

Tuba includes one PDF , DOC, RTF, PTT and XLS converters for converting file contents into text , in order to be indexed.

* 1. **Search modes :**

Besides the advanced search options :

- Search for a single word

- AND and OR search

- Search for a phrase

Tuba offers 5 additional modes to enter queries:

- Search with wildcard \*

- Strict search !

- Link search

Wildcard, strict search modes are available only for single query word input.

 **Search with wildcards \***

This mod enhances Tuba capabilities to search also for parts of a word. The mod is invoked by

entering a \* as wildcard for the unknown part of the search query.

Wildcards could be used like:

\*searchme

\*searchall\*

\*search\*more\*

Depending on Tuba keyword table, a lot more results may appear. In order not to confuse the user, the printout of relevance (weight/hits) is suppressed in result listing.

**Strict search !**

This variant is invoked by entering a !as first character of the search query. If you search for '!plus' only results for the word 'plus' will be presented in the result pages. This is the reverse function of 'Search for part of a word by means of \* wildcards'. Strict search only indicates results in the text part of the indexed pages.

 **Link search site:**

Invoked by starting the query input with ' site: ', the user is enabled to search for all pages of a domain. It is not necessary to enter the full domain address. For example if you enter 'site:maannews.net you will get a list of all pages that belong to the domain <http://www.maannews.net> If the search query is part of more than one domain address in Tuba site table, a list of these domains will be presented as intermediate result. If you then click on the desired domain of this list, all links (pages) of this domain will be presented as final result listing.

* 1. **Media search for images , video and audio :**

Media will be found individually for:

Images

Audio

Video

In image search tab : Entering 'media:' (without quotes) will present all media stored in the database.

Index of media files is performed for all media of

- Images

- Audio streams

- Videos

Three separate files in subfolder .../include/common/ that are named

image.txt

audio.txt

video.tx

hold a list of associated file suffixes. Only media files with the corresponding suffix will be taken into account during index / re-index procedure. These three files may be edited for personal purpose. In order to be indexed, for images additionally the minimum width and height (H x V pixel) is specified. Image size will be observed for the following image types:

.bmp .gif .j2c .j2k .jp2 .jpc .jpeg .jpeg2000 .jpg .jpx .png .swc .tif .tiff .wbmp

Indexing also allow selecting whether embed and nested media files should be indexed. This was implemented, as some server hide their media files as embedded objects. Another indexing procedure is used to enable indexing of external media content. When linked by the currently indexed page, also external hosted media files will be indexed. This setting is independent from the 'Tuba can leave domain', which is used for text links only. Depending of the installed GD-library, during index / re-index procedure Tuba will create thumbnails for the following image types: gif, png, jpg, ipeg, jif, jpe, gd, gd2 and wbmp

Thumbnails will be created as 'gif' or 'png' files. the gif files do have a lower quality, but will reduce the required memory for about 50%. Re-sampling the original images, size of thumbnail is defined to a maximum of 160 x 100 pixel. In result listing these stored thumbnails are used as preview. As far as available the Meta data ID3 and for images EXIF information is indexed and herewith become searchable.

In order to create thumbnails and to index ID3 and EXIF information, it is necessary to download the media file. For pages with multiple media content, the time for index /re-index procedure may increase dramatically. As ID3 information is not available for all audio and video files, the minimum play time in order to be indexed was not yet implemented .Tuba does not store the media content. Only the links, thumbnails and Meta information are stored.

**Not supported media content**

The following examples demonstrate the currently existing limitations for media data that will not be indexed:

1. If inserted in documents like pdf, doc, ppt, etc.

2. If inserted in Java or applets like:

<P><OBJECT classid="java:program.start"></OBJECT>

and also direct applet implementations like:

<APPLET code="Bubbles.class" width="500" height="500">

Java applet that draws animated bubbles.

</APPLET>

3. Image maps that are server-side or client-side included like:

<P><A href="http://www.acme.com/cgi-bin/competition">

<IMG src="game.gif" ismap alt="target"></A>

**Search for media contents :**

Tuba has search categories for all media such as images , video and audio

Each section will present result number, media title and the page address (link) at which the media was found. The image result section presents a thumbnail, the image size (H x V pixel) and a link to EXIF , information for each found image. Clicking on the thumbnails will open the original image in a new window / tab.

Video and audio results are presented with title, link to ID3 information. Media content will be opened with the belonging software by clicking on the media title.

* 1. **RSS and atom feed :**

content of RSS and Atom feeds will be indexed / re-indexed. The following content is indexed and herewith becomes searchable after indexing:

- Channel/Feed: Title and Description.

- Item: Titel, Description, Guid, Author, Category, Publication date and time.

RSS and Atom feeds are treated as normal text pages. The suggest framework will offer keyword proposals.

Also pre-selection of categories is taken into account. Feed links are treated like the standard page links.

Finally I have the file search mode in which user can search for any file of types : pdf , doc , xls and ppt

**Resources :**

<http://www.seobythesea.com/?p=2795>

<http://www.howstuffworks.com/search-engine.htm>

[www.robotstxt.org](http://www.robotstxt.org/)

[searchenginewatch.com](http://searchenginewatch.com/)