Webinator WWW Site Indexer Version 4.1

Thunderstone Software

July 29, 2002

Contents

Chapter 1

Document Conventions

Webinator will run on Windows NT, Windows 2000, and Windows XP. This document will refer to all versions of Windows as simply Windows.

Webinator runs on many versions of Unix and Unix like operating systems. This document will refer to all variations as simply Unix.

All filesystem and URL paths will be based on the default installation location. `INSTALLDIR` will sometimes be used to indicate the directory into which you installed Webinator. The default location for Unix is `/usr/local/morph3`. The default location for Windows is `c:\Program Files\Thunderstone Software\Webinator`.

Examples of command lines and URLs may be broken into multiple lines to fit the printed page. You should not split them when entering them at a command prompt. The split is indicated by ⇝ at the end of the printed line and ↪ at the beginning of the next printed line.

```
http://www.somesite.com/this/a/long/url/with/many/⇝
   ↪subdirectories/that/won't/fit/on/a/line.html
```

If a space is required between the two portions it will be indicated with □.

```
INSTALLDIR/bin/texis profile=PROFILENAME□⇝
   ↪DOCUMENTROOT/webinator/dowalk/dispatch.txt
```

Chapter 2

Overview

Webinator is a web walking and indexing package that allows a web site administrator to provide a high quality retrieval interface to collections of HTML and other documents. It is an application of Texis and is written in Texis's Web Script language called Vortex.

It consists primarily of the Texis binary program and three Vortex scripts which are run by the Texis CGI program on your web server and are accessed from a web browser.

One script provides the administrative interface, another provides the site walker and indexer, and the third provides the search function that end users see. Since these are all scripts, they are easy to modify to provide the look and feel of your site, or to create custom rules for indexing your site.

2.1 Features

Here are some of its features:

- One or more web sites may be indexed into a single database.
- Multiple databases may be maintained.
- Support for cookies.
- Support for meta data.
- Support for proxy servers.
- Robots.txt and meta robots are respected.
- Totally customizable search interface.
- Totally customizable site walker/indexer.
- A web site may be copied to the local file system.

There are many more features and options to tailor Webinator's behavior to your needs. Almost any option not provided directly by the administrative interface may be achieved by editing the included script(s).

2.2 Obtaining Webinator

Webinator may be obtained from `http://www.thunderstone.com/webinator/`. There you may review the different versions that provide varying size limits and levels of support. Then you may download the free version or order one of the paid versions.

Follow the instructions on the web site to acquire the package for your operating system. After registering for the free version you will be given a URL to a compressed tar file for Unix versions, or a setup exe program for the Windows version, containing binaries for your specified operating system.

2.3 Technical Support

Support for Webinator is via a searchable web message board. It is located at the following URL:

`http://thunderstone.master.com/texis/master/search/msgboard.html`

Anyone may read the discussions. To post a question or comment you must create an account, which is free, and be logged in. Also, once you are signed up you may "subscribe" to periodic email notifications of new postings to the board. You may select hourly, daily, or weekly notification of new postings.

If you subscribe to periodic notifications, and at some point in the future no longer wish to receive them, you may select "subscribe" again to enter the administrative area where you may delete your subscriptions.

Do NOT attempt to get support for free Webinator by any other email or voice channel. Paid users may also use the "Tech Support" form at

`http://www.thunderstone.com/`

Other Webinator resources, such as FAQ, alternate search examples, and such may be found at Webinator's home page `http://www.webinator.com/`.

Chapter 3

Installation

3.1 Unix Download and Installation

For Unix platforms, download the `webinator-4.1.tar.gz` file, from the URL given to you during the registration procedure, to a temporary directory on your machine. (The number `4.1` in the filename may differ, if you are downloading a different version.) Then uncompress it, extract it, and run the install script:

```
gunzip <webinator-4.1.tar.gz | tar xvf -
sh ./install
```

Note: The Webinator install should preferably be run as the user that will actually run the software, not as `root`. The user should be the same user that your web server runs CGI programs as (typically a non-login user); consult your web server config files for details. This user must have permission to place and move files in the install directory and the web server tree. If you must run the install as `root` it will ask you for the name of a non-`root` user that Webinator will run as. **Note:** Once installed, Webinator should **never** be run as `root`.

You will be asked several questions during the installation. For some of these questions, a default answer may appear in square brackets. Eg.:

```
Install dir [ENTER for /usr/local/morph3]:
```

In this case, if you just hit `Enter` without typing a path, the install will use the answer `/usr/local/morph3` as if you'd typed that. **Note:** Just because a default answer is given, does **not** necessarily mean that is the correct or best answer for your particular environment. It is up to you to choose the default or enter your own value based on knowledge of your machine's setup.

The questions you will be asked include:

- **Install directory**
  This is the directory where Webinator will place its files and subdirectories. It should be a unique (empty) directory; if it does not exist the install script will ask to create it for you. The standard install

directory is `/usr/local/morph3`; you should use this if at all possible to avoid potential path issues later. Only enter a different directory if you are specifically unable to install to the standard directory. Whatever directory you choose should be *in*accessible to your web server (ie. outside its server and document directories): the install will place just the public files of Webinator in your web server tree later.

- **CGI directory**
  This is the directory that your web server runs CGI programs from. The install will create a symbolic link to the `texis` executable here. **Note:** Since Webinator runs as a CGI program your web server **must** be configured to run CGI programs. Consult your web server documentation and config files to find out how and where your server places CGI programs. For Apache servers it is typically done with a `ScriptAlias` directive. Note that this is the *file* path to your CGI directory, not the URL entered in a browser.

- **CGI URL prefix**
  This is the URL prefix to the CGI directory you just entered. In other words, it's the URL that you would enter in a browser to access a CGI program in that directory, but without the program name.

  For example, let's say you already have a CGI program `findit` installed on this machine, and you access it via the URL `http://www.mysite.com/cgi-bin/findit`. You would then enter `/cgi-bin` as your URL prefix. If your site uses virtual hosts, or runs on a non-standard port, you can enter a full URL instead (eg. `http://www.myothersite.com:2001/cgi-bin`).

  If you want to start over with a new CGI directory (previous question), then enter `/newdir` to back up a step.

- **CGI extension**
  This is the filename extension that CGI programs have in the URL. On some web servers, instead of just one directory for CGI programs, any program with a special extension such as `.cgi` at the end signifies a CGI program. If this is true for the CGI URL prefix you've selected, enter it here. For example, if your CGI programs are named `findit.cgi` or `shop.cgi` then you might enter `.cgi` as the extension. (This may be the case for Apache servers if CGI is set up with an `AddHandler cgi-script` directive instead of `ScriptAlias`.) If your programs do not have an extension in the URL, type `none`.

- **Webinator admin password**
  This is the password that the default Webinator administration account will have. This password is used to control access to your Webinator walks, so choose a password with care, and ensure that only authorized administrators know it. (Once installed, you can create multiple administration accounts with different passwords if you desire, from the web-based admin interface.) Under some circumstances on some OS's setting the password from the install may fail. Don't worry. You will be asked to set the password the first time you access the administrative interface.

Once the installation has completed successfully, you can remove the tar and install files, as they are no longer needed:

```
rm -f install webinator-4.1.tar.gz webinator.tar□↪
   ↪install.sum webinator.sum
```

Note: If you move your web server directories around or change your CGI configuration after installing Webinator, you will have to re-install it.

3.2 Windows Download and Installation

The Windows version of Webinator runs on NT 4, Windows 2000, and Windows XP. Download and run the installation program `webinator-4.1.exe` from the URL you were given during the registration procedure. (The number `4.1` in the filename may differ, if you are downloading a different version.)

IIS NOTES:

- A default install of IIS may not include the `scripts` virtual directory. Before proceeding with the install make sure that the `scripts` virtual directory exists, or that another directory has been created with *Execute Permissions: Scripts and Executables*

- The URLs to use Webinator will include `/texis.exe`, so if you have installed *URLScan* you will need to allow `.exe` extensions.

During the install you will be prompted for the following locations:

- **Install directory**
  This is the directory where Webinator will place its files and subdirectories. The directory you choose should be *in*accessible to your web server (ie. outside its server and document directories): the install will place the public files of Webinator in your web server tree later.

- **CGI directory**
  This is the directory that your web server runs CGI programs from. **Note:** Since Webinator runs as a CGI program your web server **must** be configured to run CGI programs. Consult your web server documentation and config files to find out how and where your server places CGI programs. Under IIS it requires *Execute Permissions: Scripts and Executables* permissions. Note that this is the *file* path to your CGI directory, not the URL entered in a browser. If you are using IIS the install will attempt to find a suitable directory. A typical default would be `c:\inetpub\scripts`

- **HTML directory**
  This is the directory that your web server gets HTML pages from. Consult your web server documentation and configuration to find out how and where your server looks for HTML files. The install will create a directory called `Webinator` and install the publicly visible files, such as the search form and graphics. If you are using IIS the install will attempt to find a suitable directory. A typical default would be `c:\inetpub\wwwroot`

- **Webinator admin password**
  This is the password that the default Webinator administration account will have. This password is used to control access to your Webinator walks, so choose a password with care, and ensure that only authorized administrators know it. (Once installed, you can create multiple administration accounts with different passwords if you desire, from the web-based admin interface.)

3.3 Filesystem Layout

Webinator is installed underneath `/usr/local/morph3` on Unix or `Program Files\Thunderstone Software` on Windows by default. It consists of several subdirectories. This will be the structure on Unix (not all files are listed here):

```
Install Directory
   Readme.txt
   license.key
   texis.cnf.sample
   .htaccess
   webinator/
      newindex.html
      dowalk
      webinatoradmin
      search4
      webinator4man.pdf
      swdrmlog.gif
   bin/
      texis
      monitor
      anytotx
      gw
   texis/
      monitor.log
      vortex.log
      testdb/
      default/
         db1/
         db2/
HTML Directory
   webinator/
      index.html
      dowalk
      webinatoradmin
      search
      webinator4man.pdf
      swdrmlog.gif
CGI Directory
   texis
```

Note 1: The files under `webinator` are also installed into the `webinator` directory under your specified document root. `newindex.html` will be named `index.html` if an old copy was not already present. `search4` will be named `search` if an old copy was not already present.

The `webinator` directory contains the search interface scripts, several GIF files used by the search

interfaces, and an `index.html` that contains a hyperlink to the administrative interface, as well as the online documentation.

All of the directories that should not be referenced by web browsers contain a `.htaccess` file that denies all access in the event that you choose to install under your web server's document tree. If you installed under your document root and your web server does not respect `.htaccess` style protection you should block web access to those directories by whatever means your web server provides.

This will be the structure on Windows (not all files are listed here):

```
Install Directory
   license.key
   texis.cnf.sample
   texis.exe
   monitor.exe
   anytotx.exe
   gw.exe
   texis\
      monitor.log
      vortex.log
      testdb\
      default\
         db1\
         db2\
HTML Directory
   webinator\
      default.htm
      dowalk
      webinatoradmin
      search
      webinator4man.pdf
      swdrmlog.gif
CGI Directory
   texis.exe
```

The `bin` or `Install` directory contains the `texis` program and other related utility programs. The `gw` program from version 2.5 Webinator is included in this transitional release. It may go away in future releases. It will work with existing Webinator 2.5 databases. **Note:** `gw` will **not** work with new Webinator 4 databases.

The `texis` directory contains the databases and Texis log files.

3.4 File Permissions and OS Specific Notes

- **Windows**
  IIS will typically run `texis.exe` as the anonymous user `IUSR_machine`. If you want searches to

automatically recompile scripts for you then this user will need write permission on the directories containing the scripts.

Another option is to test and compile the scripts in a staging area, and when you are satisfied with the results simply move the compiled `.vtx` file into place.

Texis requires that its monitor process be running. It will attempt to start it if it's not already running. When Texis is running under the web server there may not be permission available for it to run properly. You can, as administrator, register the Texis monitor as a service to run in background and when the system starts up. The install will do this if run as an administrator. You can do this manually from a command prompt when logged in as administrator:

```
monitor -R
```

This will start the monitor service immediately so there's no need to reboot to activate it.

If you ever wish to unregister the Texis monitor as a service do this from a command prompt when logged in as administrator:

```
monitor -U
```

- **Unix**
  It is important that texis and its related utility programs always run as the same userid and that that userid is the owner of the databases. Web servers generally run CGI programs as some user with little or no permission. The installation attempts to get around this problem by making the programs `setuid` to the correct user. If it is not able to you will receive a warning. It is up to you to ensure that texis is always run as the same userid.

  The standard Unix commands for making a program `setuid` to some user, say `myself` for example, are:

  ```
  chown myself texis
  chmod u+s texis
  ```

  The above commands may only be run by the `root` user on some systems.

3.5 Customizing Webinator's Appearance

You may make common changes to Webinator's search appearance by using `Search Settings` from the administrative interface main menu. You may select color, font, size, result style and order, as well as setting boilerplate HTML to wrap around the search form and results.

But you are not limited to these features. You may change any and all aspects of the search program's appearance and behavior by modifying the supplied `search` script or writing an altogether new one.

See `http://www.thunderstone.com/webinator/example/` for some examples of custom scripts.

For details on programming with Texis Web Script (Vortex), see the manual at the Thunderstone web site, `http://www.thunderstone.com`.

See also `Customizing the Search` 6.3 for some insight into the inner workings of the default search script.

Chapter 4

Operation

4.1 Running the Administrative Interface

Note to Webinator 2 users: The Webinator 2 `gw` *command is included in the Webinator 4 package. See "Using gw" (4.7.2) for details.*

The administrative interface to Webinator is a web application that you access using your web browser. Access it using the URL that was given to you during installation. It will be something like:

On Unix: `http://YOURSERVER/cgi-bin/texis/webinator/dowalk`

On Windows: `http://YOURSERVER/scripts/texis.exe/webinator/dowalk`

Where `YOURSERVER` is the hostname, and possible port number, used to access the web server where Webinator is installed.

The `cgi-bin` and `scripts` portions refer to the CGI directory you specified during installation. The examples given above are the most common. Your path could be different.

`texis` and `texis.exe` are the names of the Texis Web Script interpreter and is a program that resides in your CGI directory. It is not a directory.

The portion after `texis`, `/webinator/dowalk`, is a "virtual" path indicating the location of the administrative script relative to your web server's "document root" directory. The administrative script is called `dowalk` and is installed in the directory called 'webinator' under the document root you specified during installation.

When you run the administrative interface you will be asked for the login and password. By default there is one login name. It is `webinator` in all lowercase. If no other accounts have been added you will not have to enter the name. It will be filled in for you. Your login will be remembered in a cookie until you logout. This way you don't need to enter the password every time you enter. **Note:** If you share your computer or it is otherwise available to people that should not be administering the Webinator system you should logout when you are finished.

The administrative interface uses JavaScript to enhance its functionality and ease of use, but it will also work well without it. No Webinator functionality will be lost without JavaScript. In this document the user

interface will be described assuming that JavaScript is enabled.

4.2 First Time Run

During installation you were asked for a password for the default administration account (`webinator`), which you should now enter at the prompt. If for some reason this step did not happen, the first time you run the administrative interface you will be asked to create and enter a password. You should choose a password that is easy for you to remember but hard for someone else to guess. You will need to enter the same password twice (two input boxes will be provided) to protect against typing mistakes. Passwords are case sensitive.

Once you create the password you will be automatically logged in and shown the `Choose a profile` page. A default profile name and data directory will be filled in for you. You may change either of these if desired, then hit the `Create Walk` button. A new profile will be created but a site walk/index will not be started yet.

You are then presented with the main walk settings page. The `Base URL` will be automatically filled in with the name of your web server. If you wish to walk a different site you may change the `Base URL` at this point.

If your site has pages that you want indexed with extensions other than `.html`, `.htm`, or `.txt` you should add them to the `Extensions` list. Also note that extensions are case sensitive unless you use `Ignore case` under `All Walk Settings`.

Once you're happy with the URL and extension settings you may hit the `GO` or `Update and GO` button to begin a walk of your site. A walk will be started in the background and you will be taken to the `Walk Status` page. This page will show you the status of the walk in progress and indicate when the walk is complete. This page will automatically refresh every 15 seconds with the latest progress information until the walk is complete. When the walk is complete you will see a summary of errors, if any.

Once the walk is complete you may click `Live Search` on the menu at the top of the page. This will take you to the search that users will use. It is also the URL you can place on your web page(s) to send users to the search.

You now have a site index that you can use. There are many options to control the site walk as well as the search interface appearance. They are described in detail elsewhere in this manual. Use the `All Walk Settings` button on the administration script's menu to see all of the options. Click the question mark (`?`) next to an item to get help for that item.

Since the walker, administrative interface, and search are all scripts with source provided you are not limited to the settings available in the administrative interface. Any or all of the scripts may be modified to take on new behaviors.

4.3 Administrative Interface Overview

Webinator's administrative interface menu has the structure given below. Each item will be described on the pages that follow.

```
Entry
   Basic Walk Settings
      Update
      GO, Update and GO
      STOP
   All Walk Settings
      Update
      GO, Update and GO
      STOP
   Search Settings
      Update
   List/Edit URLs
      Live Search Database
      New Walking Database
   Walk Status
      Refresh
      STOP Walk
   Query Log
   Test Search
   Live Search
   Profiles
      Create a Profile
      Select a Profile
      Delete a Profile
   Accounts
      Add a User
      Change Password
      Delete
   Documentation
   Webinator Home
   Logout
```

4.3.1 Entry

Upon entry to Webinator administration you will be asked for user name and password. If you have logged in previously and still have the cookie and have not logged out the login page will be bypassed and you will be taken directly to `Profiles` (see section 4.3.13).

Your login will be remembered in a cookie until you logout. This way you don't need to enter the password every time you enter. If you share your computer or it is otherwise available to people that should not be administering the Webinator system you should logout when you are finished.

4.3.2 Basic Walk Settings

This is the central area for configuring a walk. The most commonly used walk related options and their settings are enumerated and may be changed here. Next to each option is a question mark (?) which, if clicked, will take you to help for that option. The options are documented individually later in this manual in section 4.4.

At the bottom of the page is a set of three buttons. Pressing any of the buttons affects all options on the entire page.

- `Update`
  This button causes all changes on the form to be saved. No walk is started.

  If the rewalk schedule has been changed the new schedule will go into effect immediately.

  If categories have been changed the walk database will be updated to reflect the new categories. The search interface will then reflect the new categories.

  If single page, page file, or page URL have been changed, the listed individual pages will be fetched into the live search database and made available for searching.

  If the word definition is changed, the search index on the live database will be dropped and recreated. Searches may not work while the index is being rebuilt.

- `GO` or `Update and GO`
  The `GO` button will change to `Update and GO` after you make a change to any setting on the form. The ultimate behavior for either is the same.

  The current settings from the form will be saved as when you hit `Update`. Then a new walk will be started. The new walk will be performed to a temporary database so that the live search is not disturbed at all by the new walk. Then you will be shown the walk status page where you may monitor the progress of the walk.

  Changes to categories or word definitions will not be reflected until the walk finishes.

- `STOP`
  When a walk is in progress the `GO` button will be replaced by the `STOP` button. This button will terminate the running walk and abandon the work that it has done so far.

- `Reset`
  This button reverts all settings on the page to what they were when the page was first loaded.

4.3.3 All Walk Settings

This is the central area for configuring a walk. This is the same as `Basic Walk Settings` except that all walk related options and their settings are enumerated and may be changed here.

4.3.4 Search Settings

This page contains all of the settings related to the search interface that end users see when performing searches.

All search options and their settings are enumerated and may be changed here. Next to each option is a question mark (?) which, if clicked, will take you to help for that option. The options are documented individually later in this manual in section 4.6.

At the bottom of the page is a set of two buttons. Pressing any of the buttons affects all options on the entire page.

- `Update`
  This button causes all changes on the form to be saved.

  Changes made to the appearance options will be immediately visible in the test search. If apply appearance is checked they will also be immediately visible in the live search.

- `Reset`
  This button reverts all settings on the page to what they were when the page was first loaded.

4.3.5 List/Edit URLs

From here you may list or delete all or selected URLs from the database. (You should always list before you delete so that you know that you are deleting the correct ones.) While listing URLs you may display all known information about a given page. You may also create categories for selected sets of URLs from this interface.

If a walk is in progress delete will be disabled and you will be given the choice of listing URLs from the live search database or the new database being built by the walk.

Select `List` or `Delete` from the drop down list. The default is always `List` for safety.

Enter the URL or pattern for URLs that you want information about in the pattern box. This may be an exact URL or a wildcarded pattern to list all URLs matching the pattern. Use asterisk (`*`) to match anything and question mark (`?`) to match any single character. You may enter up to 10 different URLs or patterns in the box to find them all at once. Put a space between patterns when entering multiples. Leaving the pattern box blank implies `*` and will cause every URL in the database to be listed. Deletion will be denied if the pattern is blank or `*`.

Select the order in which you wish to see the list:

Depth	URLs encountered first in the walk will be listed first
URL	URLs are ordered alphabetically
Newest first	URLs are ordered by modification date with newest ones first
Oldest first	URLs are ordered by modification date with oldest ones first
Largest first	URLs are ordered by download size with largest ones first
Smallest first	URLs are ordered by download size with smallest ones first

Then `Submit`.

All matching URLs will be listed. Clicking on a listed URL will take you to a page of details about that URL. On that detail page you will see everything the database knows about that URL. You will also be able to see what pages refer to the selected page by clicking on `Parents` and what pages the selected page refers to by clicking on `Children`.

If your pattern matches less than the entire database you will be given a form from which you can create a category using the same pattern(s). Simply enter the name of the category to create and hit `Submit`. The name is the name that users will see on the search form. This new category will also appear on the main settings page along with the other categories. It will also be immediately available to search users.

4.3.6 Live Search Database and New Walking Database

These options are presented on the `List/Edit URLs` page (see 4.3.5) if there is a walk active. They allow you to choose which database to query. The “Live” database is the one from a previous successful walk that is what search users see. The “New” database is the database currently being built by the new walk. It is not visible to search users.

4.3.7 Walk Status

This page shows the status of the latest walk for the current profile. If a walk is in progress that is the one that will be reported.

During an active walk it will indicate walk start time, starting URLs, the number of pages fetched so far, the number of errors and duplicates encountered so far, and the most recently fetched URLs. The page will automatically update every 15 seconds until the walk is complete or another page is selected. During the walk `Refresh` may be selected to force a refresh before the 15 second automatic refresh. Also `STOP` may be selected to stop the walk and abandon it.

When no walk is in progress the report will also include a list of errors and duplicates encountered.

If the last walk was abandoned, the report will include information about how far it got as well as the report from the last complete walk.

4.3.8 Refresh

During a walk use this to refresh the walk status page before the default 15 second interval.

4.3.9 STOP Walk

Use this to stop and abandon a walk in progress.

4.3.10 Query Log

The query log will display the current state of the query logging option and information about any searches performed while query logging is on. If query logging has never been turned on for the current profile there will be nothing to see. The query log is erased each time the database is rewalked.

The query log will list the time that each search occurred, the IP address of the web user performing the search, the number of hits for the search, and the user’s query. For URL clickovers it will display the query

instead of the number of hits and the actual URL instead of the query.

Selecting the Date/Time for a listed query will display a page with complete info about the search. This page includes everything from the summary list, as well as any non-default parameter settings from the search. A hyperlink is provided so that you may perform the same query as the user.

4.3.11 Test Search

This hyperlink will jump to the search interface. It will force the interface to use the search settings listed on the `Search Settings` page whether they have been applied or not. This allows you to test search settings without affecting end users until you are satisfied with the new settings.

This mode will also place two extra hyperlinks at the top of the search pages.
`Back to Administration` will allow you to return to the Webinator administration interface.
`Make this appearance live` will do that, as well as making the search settings you are testing "live" so that end users will now see them.

4.3.12 Live Search

This hyperlink will jump to the Webinator search interface as end users see it.

4.3.13 Profiles

This page will present a list of existing profiles. You may click on the profile name to see and/or change it's settings and status or to start a walk.

You may click on `Delete` next to a profile to delete that profile. You will be asked whether you really want to delete the profile or not.

When a profile is deleted all of its settings are lost and any walk database it may have created is deleted. There is no way to get any of these back once the profile is deleted. Under Windows it is possible that the walk database will not be completely deleted if there are currently searches being performed on the database. You shouldn't be deleting a database that is being actively searched. If you get into this situation you will have to delete the remnants of the database by hand.

You may also create a new profile by entering a new name and data directory. You may not use a data directory that is in use by another profile. You would generally specify a new data directory. The directory will be created if it does not already exist.

You may copy settings from an existing profile to your new profile by selecting it's name from the drop down list. This would allow you to setup another site similar to an existing one. It would also allow you to experiment with the walk settings for an existing site without potentially messing up the good walk that is being searched by your users.

You may also import options from an existing Webinator 2 database that you may have. To do this fill in the `New Profile Name` and `Data Directory` normally, then also fill in the `Webinator 2 database` field with the full path to the old database from which you would like to

import options. If the options were stored in a profile other than the default of `lastrun` using the `gw -save` fill in the name of the desired profile in the `Webinator 2 profile` box. Then click `Import Settings`. This will create a walk and load settings from the specified old database.

Notes about the import process and differences between versions 2 and 4.

- **-[no]unique**: Databases are unique by default (“Prevent Duplicates” 4.5.30).
- **-j**: It is automatically implied (“Stay Under” 4.5.29).
- **-k**: The default expressions are broader (“Word Definition” 4.5.25). The import will replace the defaults with what is your old profile.
- **-n**: All known plugins are predefined in dowalk function `doplugin`. You may need to add extensions to the “Extensions”4.4.6 list and/or mime types to the “Mime Types”4.5.41 list though. Import will do this for you.
- **-r**: Robots META tags are also supported. Import will apply your old setting to both robots.txt and meta robots.
- **-b**: Permanently on (walks are always breadth first).
- **-L**: Permanently on (virtual hosting demands it).
- **-l**: Not changeable. Log files contain different information.
- **-q**: Quit time is not supported.
- **-c, -A**: Site copying is not supported.
- **-[no]dnscache**: DNS caching is not supported.

4.3.14 Accounts

This section allows you to maintain multiple login accounts for access to Webinator administration. All users will be listed on this page. You may add users, delete users, and change individual user passwords. The default user, called webinator, may not be deleted.

It also allows you to create multiple administrative users. There is no distinction amongst them once created. All users have full administrative permissions and may create and delete any user or change any user’s password. This is a basic security mechanism meant to keep unauthorized persons from using the web based administrative interface. The purpose for multiple users is so that you can create distinct passwords that you might want to revoke in the future without having to change a single global password that all administrators know.

User names and passwords are stored in the SYSUSERS table of the default database. This is only a holding place for them. No Texis permissions are granted or revoked for these users. A side effect of the users being stored in SYSUSERS is that any users that you might create in the default database by other means than the Webinator interface will also automatically become Webinator administrators.

The passwords are forward encrypted. This means that a forgotten password may not be discovered. The only way to deal with a forgotten password is to change the password. In the event that all passwords are forgotten you can delete the webinator user from SYSUSERS using texis -s from a command prompt and entering an appropriate SQL delete statement. The administrative script will then create the webinator user anew and ask you for a new password.

4.3.15 Add a User

To add an administrative user enter the new user's login name and password. You will have to enter the new password a second time into the `Confirm` box to protect against typing mistakes (since you can't see the password you are typing).

Names and passwords are case sensitive. "Joe" is different than "joe". You should choose passwords that are easy to remember but hard for someone else to guess.

4.3.16 Change Password

Here you may change the password for the selected user. You will have to enter the new password twice to protect against typing mistakes (since you can't see the password you are typing). Enter the password once the `Password` box and again into the `Confirm` box

Passwords are case sensitive. "Joe" is different than "joe". You should choose passwords that are easy to remember but hard for someone else to guess.

4.3.17 Delete

This will delete the selected user. You will be prompted to confirm whether the user should really be deleted or not. Once a user is deleted there's no way to get it back except to re-add it.

The default user, "webinator", may not be deleted.

4.3.18 Documentation

This provides a hyperlink to the online version of this document.

`http://www.thunderstone.com/site/webinator4man/`

4.3.19 Webinator Home

This provides a hyperlink to the online home of Webinator.

`http://www.thunderstone.com/texis/site/pages/webinator.html`

4.3.20 Logout

This will log you out of the administrative interface and clear your login cookie. It will then take you back to the login page.

4.4 Basic Walk Settings

These are the settings most commonly used and are available in `Basic Walk Settings`.

4.4.1 Database

Syntax: the full path to the database directory on the server's disk

This indicates what database is being used by the currently selected profile. The database is only settable when creating a profile. A new profile must be created to use a new database.

4.4.2 Walk Summary

This is informational only. It contains summary information about the most recent walk and recategorizations.

4.4.3 Base URL

Syntax: one or more HTTP URLs, one per line

This is the address where the web crawler will start walking your site. If the whole site is to be searched, simply enter your web address, ex - "`http://www.mysite.com`". If the search is to be limited, specify the address to start the search or create a page listing the URLs to search. The search will only return information from your web site - no off-site searching will be done. Directory URLs should include a final forward slash "`/`". Example - "`http://www.somehost.com/mysite/`". If you have a virtual domain that just redirects to another URL, enter the destination URL as your Base URL instead of your virtual domain name.

You may specify multiple base URLs to index multiple sites. Webinator's idea of a "site" is a single host as identified by the hostname portion of a URL. Therefore `http://www.mysite.com`, `http://www2.mysite.com`, and `http://mysite.com` would all be considered different sites.

See also `URL file` 4.5.2, `URL URL` 4.5.3, `Single page` 4.5.4, `Page file` 4.5.5, and `Page URL` 4.5.6 for more ways to specify URLs.

4.4.4 Enterprise

Syntax: a single domain name

The name of your company's domain. This is useful if your company's web presence consists of multiple hosts within it's domain and you want them all indexed together as a unit.

This allows you to walk any URLs encountered during the walk of the base site(s) that are within the given domain. Webinator will attempt to guess this value for you but you may set it to whatever you wish. Check the `Yes` box to enable this feature.

See also `Extra domains` 4.5.9 which is the same but allows more than one domain. These options may be used together.

4.4.5 Robots

Syntax: select Yes or No buttons

robots.txt

With this set to Yes Webinator will initially get `/robots.txt` from any site being indexed and respect its settings for what prefixes to ignore. Ignoring robots.txt is not generally recommended.

See also `Robots.txt` 5.3.

Meta

Respect the meta tag called `robots`. With this set to Yes Webinator will process and respect the robot control information within each retrieved HTML page.

See also `Robots.txt` 5.3.

4.4.6 Extensions

Syntax: one or more file extensions separated by space

A list of the URL extensions that the crawler will accept. The defaults are

```
.html
.htm
.txt
.pdf
```

To search MS-Word documents, use `.doc`. For Shockwave/Flash use `.swf`. For WordPerfect documents specify whatever extension you use and ensure that the web server returns the mime type `application/wordperfect` as there is no consistent extension for WordPerfect documents. Any extensions not listed here will not be searched or walked.

A few other extensions you may find useful are

```
.asp
.jsp
.shtml
.jhtml
.phtml
```

4.4.7 Exclusions

Syntax: zero or more strings, each on a separate line

Excludes URLs containing any of the specified literal strings anywhere in the URL (hostname, path, or query).

See also `Exclusion REX` 4.5.11 and `Exclusion prefix` 4.5.12 for more ways to exclude URLs.

4.4.8 Crawl Delay

Syntax: a whole number from 0 to 10

Causes Webinator to wait the specified number of seconds between page fetches. This is normally 0 but may be increased if the web server can't handle being hit rapidly. Increasing this value will force the walk to take at least this number, times the number of pages on the site, seconds to complete.

Note: Using a delay larger than 0 will force `Threads`(4.4.9) to 1. A delay defeats the advantage of multiple threads and large delays could cause unexpected page fetch timeouts.

4.4.9 Parallelism

Syntax: whole numbers from 1 up

Threads

This is the maximum number of simultaneous page fetching threads to allow against each site. Setting higher than 5 is probably not very helpful unless you have many "Single Pages" that are on various hosts.

Servers

This is the maximum number of different web servers to walk simultaneously. Turning this up too high can stress your memory, cpu, and network.

4.4.10 Verbosity

Syntax: whole number from 0 through 4

Table 4.1: Verbosity Levels

Level	Description
0	Issue no messages except errors
1	Display starting point URLs
2	Display selected setting info
3	List URLs found in URL files
4	Indicate why URLs are rejected

Sets how much information the walker should provide about what it's doing. The default verbosity level is 2.

Each level includes the previous levels.

4.4.11 Rewalk Type

Syntax: select from drop down box

This determines how rewalks are performed. The default type is `New` which creates a new database and does a complete walk of everything while not disturbing the existing database.

The rewalk type `Refresh` works on the existing database and only downloads files that have been modified or created since the last walk. Pages that are no longer present on the server are removed from the database. Pages that were referenced but missing in the initial walk but appear later will be missed by refresh if their parent page has not been modified. If you change your settings to be more inclusive (ie add extensions, ignore robots, add domains, etc.) you should do a `New` walk once as a `Refresh` will not be likely to find the newly allowed data unless all of the pages leading to such data have been modified.

If more than 30%-50% of your site changes between walks you may be better off using a `New` walk instead of `Refresh`. Also, many dynamic content generators do not give modified dates which will cause every page to be rewalked. In that case you should use `New` instead of `Refresh`.

Method	Advantages	Disadvantages
`New`	Guarantees most accurate representation of current site.	Uses more bandwidth.
	Does not disturb an existing search database.	Uses more temporary disk space.
`Refresh`	Faster.	Could get out of sync with actual site under rare circumstances.
	Uses less bandwidth.	A lot of changed pages could substantially slow searches during the walk.
	Uses less temporary disk space.	Requires "if-modified-since" support on walked web server.

4.4.12 Rewalk Schedule

Syntax: select from drop down boxes

This performs a rewalk (the same as the GO button) on the schedule specified. The Frequency defines how often to automatically rewalk. The Hour defines at what hour to start the rewalk for daily or weekly runs.

See also Notify 4.4.14. If you are using "On Change" see also Watch URL 4.4.13.

4.4.13 Watch URL

Syntax: an HTTP URL

This is only used, and is required, if the "Rewalk Schedule" is "On Change". Webinator will check every 15 minutes for changes. If the web server supplies a modified date that will be used to determine change. If the web server does not supply modified date the text of the page will be compared against the last remembered version to determine change.

4.4.14 Notify

Syntax: an email address

If this is set a summary report will be sent to the supplied email address when a scheduled rewalk occurs.

4.4.15 Action Buttons

These buttons tell Webinator to do something now. They are as follows:

- Update: Save the current settings for future use but don't begin a walk.
- GO: Begin a walk using the current settings.
- Update and GO: Save the current settings then begin a walk using those settings.
- STOP: Stop and abandon the walk that is currently running.

See the Walk Settings section (4.3.2) for details about the operation of these buttons.

4.5 Advanced Walk Settings

These are the advanced settings that are less commonly used and are available in All Walk Settings. You are not limited to the features listed here. You may modify the dowalk script to behave however you want.

See also Customizing the Walker 6.4 for some insight into the inner workings of the dowalk script.

4.5.1 Categories

Syntax: textual name and URL pattern pairs, additional input boxes will appear as you use up the ones provided

Webinator can create searchable sub-categories that will appear in a drop down box on the Search page. Enter the name of the category on the left, and its corresponding URL pattern on the right. URL patterns may contain asterisk(*) to indicate "anything" and question mark(?) to indicate any single character. There may be more than one pattern for each category. Separate multiple patterns with space.

Table 4.2: Example Categories

Category	URL Pattern
Demonstrations	`http://www.mysite.com/demos/*`
Manuals	`http://www.mysite.com/manual/*`
Books	`http://www.mysite.com/a1/* http://www.mysite.com/b3/*`

This example would create a category named "Demonstrations" which would only search the URL "`http://www.mysite.com/demos/`" and any files under this directory, thereby creating a more concise match to the users search. The same is true for "Manuals". The "Books" category would include pages from both the "earnmoney" and "printmoney" directories. The user would now have the option to search within just these categories or the entire database. The pattern should NOT be a single page unless you want a category with a single page in it (e.g. `http://www.mysite.com/manual/index.html` would be incorrect). It should typically be a prefix for a directory that has multiple pages within it followed by an asterisk (*).

4.5.2 URL File

Syntax: the full path to a file on the web server's disk

This allows you to specify a file containing a list of site URLs to walk. This is an additional way of specifying more Base URLs 4.4.3. This file will be reread each time a new [re]walk is started.

4.5.3 URL URL

Syntax: an HTTP URL to a plain text file (NOT HTML)

This allows you to specify the URL of a plain text file containing a list of site URLs to walk. This is an additional way of specifying more Base URLs 4.4.3. This URL will be refetched each time a new [re]walk is started.

4.5.4 Single Page

Syntax: one or more HTTP URLs, one per line

Here you may specify URLs for individual pages to include in the index. These pages are fetched and stored

in the database like others but the hyperlinks on them will not be followed.

If you change this and hit "`Update`" instead of "`GO`" the added pages will be fetched immediately and added to the existing database. Pages removed from the list will NOT be removed from the database until the next rewalk.

4.5.5 Page File

Syntax: the full path to a file on the web server's disk

This may be used to specify a file containing URLs for individual pages.

If you change this and hit "`Update`" instead of "`GO`" the added pages will be fetched immediately and added to the existing database. The file itself is not checked for changes, and pages removed from the file will NOT be removed from the database until the next rewalk.

See also `Single page` 4.5.4.

4.5.6 Page URL

Syntax: an HTTP URL to a plain text file (NOT HTML)

This may be used to specify the URL for a plain text file containing URLs for individual pages.

If you change this and hit "`Update`" instead of "`GO`" the added pages will be fetched immediately and added to the existing database. The file itself is not checked for changes. And pages removed from the file will NOT be removed from the database until the next rewalk.

See also `Single page` 4.5.4.

4.5.7 Strip Queries

Syntax: select Yes or No button

Strip query strings from all URLs. Some URLs have query strings on the end indicated by a question mark (`?`). With this option set to Yes all query strings will be removed from URLs before they are processed or retrieved.

4.5.8 Ignore Case

Syntax: select Yes or No button

This tells Webinator whether to ignore case in URLs or not. The case of hostnames is always ignored but the case of paths and filenames is respected. Some web servers don't respect case and people use various random capitalizations within filenames making the same file look like different URLs.

4.5.9 Extra Domains

Syntax: one or more domain names separated by space or line break

Allow walk to fetch pages from any host in the specified domain(s). Any URL having a hostname ending in any of the specified domains will be accepted.

e.g.: Given a base URL of `http://www.mysite.com/` and extra domain `othersite.com` Webinator will walk all of `www.mysite.com` and any URLs referring to any machine in `othersite.com`.

This option is not a "restricter" but an "enabler". All hosts specified will be walked and any others that match the given domain(s) will also be walked.

Note: This option will NOT hunt down and walk every web server in the specified domain. It will simply allow walking them if a reference to them is encountered.

4.5.10 Extra Networks

Syntax: one or more IP address prefixes separated by space or line break

Allow walk to fetch pages from any host within the network specified by the numeric IP address(es).

e.g.: Given a base URL of `http://www.mysite.com/` and extra network `192.0.2` Webinator will walk all of `www.mysite.com` and any URLs referring to any machine having an IP address prefix matching `192.0.2`.

Note: This option will NOT hunt down and walk every web server in the specified network. It will simply allow walking them if a reference to them is encountered.

Note: Using this option has the potential to slow the walk since every URL's hostname must be looked up. If there are many different off site hosts, or your DNS is slow, the walk may be slowed substantially.

4.5.11 Exclusion REX

Syntax: zero or more REX expressions, each on a separate line

Excludes URLs matching any of the specified REX expressions anywhere in the URL (hostname, path, or query).

Table 4.3: Exclusion REX examples

REX	Matches
`/scratch[0-9]/`	a subdirectory with a name of "scratch" followed by a single digit
`[^\alnum]test[^\alnum]`	the word "test" (but not retest or tester etc.)

See also `Exclusions` 4.4.7 and `Exclusion prefix` 4.5.12.

4.5.12 Exclusion Prefix

Syntax: zero or more URL prefixes, each on a separate line

Excludes URLs beginning with any of the specified prefixes. The entire URL (hostname, path, and query) is used for comparison.

Examples:

```
http://www.mysite.com/scratch0/
http://www.mysite.com/scratch1/
http://www.mysite.com/books/t
```

See also `Exclusions` 4.4.7 and `Exclusion REX` 4.5.11.

4.5.13 Max Page Size

Syntax: a whole number from 1 up

Sets retrieved page size limit to the specified number of bytes. Pages larger than the limit will be truncated - not discarded.

Note: PDF files tend to be very large for the amount of text contained within them. Truncated PDF files are not processable due to their design. Make sure this setting is large enough to handle the largest PDF file you want to index.

4.5.14 Max Pages

Syntax: a whole number from -1 up

Limits the number of pages retrieved in a run to the specified number. Use `-1` for no limit.

4.5.15 Max Bytes

Syntax: a whole number from -1 up

Limits the number of bytes retrieved in a run to the specified number. Use `-1` for no limit. The actual limit is rounded up to include the size of the last page so that it does not get truncated.

4.5.16 Max Depth

Syntax: a whole number from -1 up

Limits the depth of page retrieval to the specified number. Use `-1` for no limit. Depth is determined by counting how many links were traversed to reach a particular page. The base URLs are all at depth 0. URLs referred to by the base URL are depth 1, and so on.

4.5.17 Page Timeout

Syntax: a whole number from 1 up

Causes Webinator to timeout after the specified number of seconds during each page fetch. This includes the time to lookup the IP address of the host, make the connection to the server, and download a single page. A time out will not cause the entire process to quit. That page will just be skipped and considered unavailable.

4.5.18 Meta Tags

Syntax: zero or more meta tag names, each on a separate line

This option tells Webinator to look for the specified meta data in fetched documents and store it in the database. This data will then be included in text searches. The meta tags “Description” and “Keywords” need not be specified here as they will be indexed by default. See below.

4.5.19 Standard Meta

Syntax: select Yes or No button

This option indicates whether or not to automatically extract the standard meta tags “Description” and “Keywords” from HTML documents. If “Yes” description and keywords meta data will be extracted and stored in their own fields within the database. Unlike other meta data which will be collected and placed together into a single meta field in the database. These meta tags will be included in the search with a higher precedence than other meta tags.

4.5.20 All Meta

Syntax: select Yes or No button

Extract all meta data from HTML documents and place into the meta field for searching. This eliminates the need to know the name of all possible meta tags. But it also opens you up to the possibility recording all manner of nonsense meta data.

4.5.21 Keep HTML

Syntax: select Yes or No buttons

Specifies whether or not to include the named type of text in the database.

ALT text

`ALT` text from `IMG` or `AREA` tags.

<**STRIKE**>

Text between and `<STRIKE>` and `</STRIKE>` tags.

<**DEL**>

Text between and `<DEL>` and `</DEL>` tags.

<**FORM**>

Text between and `<FORM>` and `</FORM>` tags.

4.5.22 Remove Common

Syntax: select Yes or No button

This will cause common leading and trailing text from pages to be removed from the database. This is good for eliminating navigation menus and other static boilerplate text at the beginning and/or end of each page.

4.5.23 Ignore Tags

Syntax: one or more pairs of strings, additional input boxes will appear as you add string pairs

All data between the specified begin and end will be stripped from the HTML before the text is extracted. These are simple strings, not patterns or REX's and the case is ignored. This is useful for excluding boilerplate or otherwise unwanted portions of HTML documents.

4.5.24 Keep Tags

Syntax: one or more pairs of strings, additional input boxes will appear as you add string pairs

All data NOT between the specified begin and end will be stripped from the HTML before the text is extracted. These are simple strings, not patterns or REX's and the case is ignored. This is useful for extracting prime interest areas of HTML pages without the surrounding boilerplate.

4.5.25 Word Definition

Syntax: one or more REX expressions, each on a separate line

Sets the word matching expression(s). Each line is a REX expression defining what is considered a word within the textual content of the retrieved documents during the index process. The default expressions will index normal words and some special items such as domain names.

You may supply multiple expressions, one per line, if you can't define your idea of all possible words in one expression.

e.g.: `>>\alpha=\alnum{1,20}` will index "words" beginning with an alphabetic character followed by 1 to 20 alphabetic or numeric characters.

Changing the word definition with `Update` instead of `Update and GO` will cause the existing search index on the data to be dropped and rebuilt. The database will not be searchable during the short time that the index is being rebuilt.

4.5.26 Login Info

Syntax: name and password

Specify a username and password for sites that require logging in to view certain pages. These are used with standard HTTP style authentication. Other proprietary authentication methods are not supported. Without proper login info protected pages will be skipped.

If you are trying to walk a site where a login form is provided on a web page you may be able to walk it by using the action URL from the form with the form variables encoded onto the end as your base URL. For example if the form variable names were Uname and Upass and the action url was `http://www.mysite.com/login.asp` you may be able to use a url like `http://www.mysite.com/login.asp?Uname=YOURNAME&Upass=YOURPASSWORD`

Note: The search interface displays hit context and has an option to view the entire text of the page. This will allow search users to view "protected" pages without entering a password.

4.5.27 Proxy

Syntax: the full URL to a web proxy server

This specifies the url (not just hostname) of a proxy web server through which to pass page fetch requests. Blank means don't use a proxy.

4.5.28 Off-Site Pages

Syntax: select Yes or No button

Allow grabbing of individual off-site pages. By default `webinator` will not retrieve pages that are not on the same host as the base URL(s). With this option pages not on the same machine will be retrieved, but none of the pages that they reference will be. This option also allows off-site redirects, frames, and iframes to be fetched.

4.5.29 Stay Under

Syntax: select Yes or No button

When this flag is yes walks will stay under the directory specified in the base url(s). When this is no the walk may wander out of the directory to other locations on the same site if a hyperlink for such a place is encountered. In neither case will the walk wander out to other sites unless they are in the list of walk URLs or allowed domains or networks.

4.5.30 Prevent Duplicates

Syntax: select Yes or No button

This option will enable extra checking for duplicate documents. Documents with the same content will only be stored once, even if their URLs are different. This is accomplished by hashing the textual content of the page and not storing any page with a hash code that is already in the database.

4.5.31 All Extensions

Syntax: select Yes or No button

Retrieve all files instead of only those listed in `Extensions`. This will turn off checking of URL extensions. All URLs will be retrieved including images and such.

4.5.32 Store Refs

Syntax: select Yes or No button

Controls whether or not URLs referenced by retrieved pages are added to the refs table. This can save some time during the walk, as well as, disk space if it's turned off. But turning it off will prevent the "`Show Parents`" option in the search from working. It will also reduce the detail available from walk error reports.

4.5.33 Inline Iframes

Syntax: select Yes or No button

This indicates whether to treat iframes as a part of the page they are on or as separate stand alone pages. Selecting yes will will make them part of the page. Selecting no will make them separate.

4.5.34 Max Frames

Syntax: a whole number from 0 up

This indicates the maximum number of frames allowed on a page. Pages with more frames than this will be discarded. If this is set to 0 the frames of framed documents will be treated as independent stand alone pages.

4.5.35 Execute JavaScript

Syntax: select Yes or No button

Execute JavaScript contained on fetched pages that might alter or generate the page content and URLs.

4.5.36 Fetch JavaScript

Syntax: select Yes or No button

Fetch JavaScript that resides in a separate URL instead of being inline on the page.

4.5.37 Max Redirects

Syntax: a whole number from 0 up

This indicates the maximum number of redirects that will be followed when attempting to retrieve a page.

4.5.38 Index Name

Syntax: one or more filenames separated by space

Set the filename assumed for directory URLs. The default is "`index.html`" and "`index.htm`". This filename will be removed from stored URLs to prevent redundant fetches of the page. So the URLs "`http://www.mysite.com/fun/`" and "`http://www.mysite.com/fun/index.html`" will be considered the same and only be fetched once (as `http://www.mysite.com/fun/`).

4.5.39 DNS Mode

Syntax: choose from drop down list

This controls how Webinator looks up IP addresses for hostnames. "`Internal`" uses Texis's own internal parallelizing name lookup routines. "`System`" uses the standard system routines. You should use "`Internal`" unless it causes compatibility problems.

4.5.40 User Agent

Syntax: full user-agent string

Set the User-Agent (browser type) to report to web servers. Normally Webinator reports itself as Mozilla version 4.0. Modify this setting to report as a different user agent.

4.5.41 Mime Types

Syntax: one or more acceptable mime types, each on a separate line

These are the mime types that Webinator will tell the web server are acceptable. Mime types have the syntax `type/subtype`. Either `type` or `subtype` may be `*` to mean "any". By default all text types are allowed (`text/*`), as well as, the most common word processor and presentation formats.

4.6 Search Settings

This group of options applies to the standard search and provide a convenient way to make common changes to the search behavior and appearance. You are not limited to the features listed here. You may modify the search script to look however you want and to behave however you want.

See also “Customizing Webinator’s Appearance” 3.5.

4.6.1 Query Logging

Syntax: select Yes or No button

This indicates whether or not the search should log user queries. If yes, users’ queries will be logged to the querylog table of the database. The contents of this table may be viewed from the `Query Log` menu of the administrative interface.

Note: The query log table gets erased on every rewalk. You will only be able to view queries that have occurred since the latest walk.

4.6.2 Result Order

Syntax: select Relevance or Date button

This determines the default ordering of search results. By default answers are ordered by rank (or relevance). Selecting “`Date`” will make search results ordered by date descending (newest first) by default. Search users may select the alternate ordering from this default.

4.6.3 Results Style

Syntax: choose from drop down list

This controls the style used for displaying individual answers to user queries. There are various styles to choose from. The arrangement and amount of information varies in every style. In the administrative interface you may click the question mark (`?`) next to “`Results Style`” to see a sample of all of the available styles.

4.6.4 Results Width

Syntax: a whole number or a percentage valid for an HTML <TABLE> WIDTH

This controls the width of the `<TABLE>`s used in the search results. This may be a number indicating a fixed width or a number from 1 to 100 followed by a percent sign(%). This tells the user’s web browser how wide to make the table.

4.6.5 Box Color

Syntax: a color name or number valid for HTML color specification

This controls the color of the “gray” informational boxes at the top and bottom of search results pages.

4.6.6 Font

Syntax: a font name valid for HTML <FONT> specification

This specifies the font to use throughout the search interface.

4.6.7 Top HTML and Bottom HTML

Syntax: HTML

This is static HTML to place at the beginning and ending of every search page respectively. It is useful for setting styles and displaying navigation menus and otherwise making the search pages look like the rest of your site.

Top and Bottom HTML when placed together should be exactly what is required to create a complete and valid HTML page. You can use your favorite HTML editor to create a page with a placeholder for the search form and results. Then cut and paste the section of HTML before the placeholder into the Top HTML and the section of HTML after the placeholder into the Bottom HTML.

If `$query` occurs within these fields it will be replaced by the user’s query.

4.6.8 Enable Sherlock

Syntax: select Yes or No button

This tells the search to include comment tags in the results page that will allow Sherlock to process the list.

Sherlock is a metasearch tool for Macintosh computers.

4.6.9 Apply Appearance and Revert Appearance

Syntax: select checkbox

Changes made to the search settings are not normally immediately visible to end users. They may be tested using the “`Test Search`” menu item. This allows you to see the effects of your changes before committing to them.

Selecting `Apply Appearance` will cause the settings currently shown on the form to be made live so that end users will see them. Once this is done there is no going back without editing the settings. There is no undo.

Selecting `Revert Appearance` will cause the unapplied search settings to be discarded. The settings on the form will be reset to those being used on the live search.

4.7 Running the Walker by Hand

4.7.1 Using dowalk

Normally a walk is initiated from the administrative interface. There may, however, be times when it is desirable to start a walk by hand from a shell (or command) prompt or as a part of some other automated task. When the administrative interface starts a walk it shows you the command line to use *(using* `gw` *is discussed later in this section).* It is of the form

```
texis profile=PROFILENAME dowalk/dispatch.txt
```

You may also specify the parameter `ttyverbose` to be 1, or higher, to tell dowalk to print various status messages to the screen when being run by hand. The form would be

```
texis profile=PROFILENAME ttyverbose=1 dowalk/dispatch.txt
```

Where `PROFILENAME` is the name of the profile you have configured using the administrative interface. You will need to supply the full path to texis if it is not in your PATH. You will also need to supply the path to the dowalk script if it is not in the current directory when you run the command.

```
INSTALLDIR/bin/texis profile=PROFILENAME□↪
   ↪DOCUMENTROOT/webinator/dowalk/dispatch.txt
```

or

```
INSTALLDIR\texis profile=PROFILENAME□↪
   ↪DOCUMENTROOT/webinator/dowalk/dispatch.txt
```

Where DOCUMENTROOT is the web document root that you specified during installation.

The walker will behave the same as it does from the administrative interface. Walk info will be logged to the same files. See section 6.1.

There are several other ”entry points” that can be used to get various different behaviors when starting the walker. They all take the same form as `dispatch` above except that `dispatch` is replaced by the name of the entry point. The entry points are:

- `dispatch`
  Start a complete new walk.
- `stop`
  To stop and abandon a walk that is in progress.

- `ifmodified`
  Checks the `Watch URL`. If the watched page has changed a walk is started. If not no action is taken. This is generally used on a frequent schedule to automatically rewalk a site if it changes.
- `singles`
  Fetches and indexes any single pages specified in the profile that are not yet in the database. You would call this after adding adding to `Single Page`, `Page File`, or `Page URL`.
- `refresh`
  Start a "refresh" walk. This walk will check all pages already in the database and download only changed ones. Missing pages will be deleted. New pages discovered on modified pages will be added.
- `recat`
  Recategorize the database based on the current settings of `Categories`.
- `reindex`
  Drop and recreate the Metamorph index on the html table. This would be used after changing the `Word Definition` expressions.
- `remakeindex`
  Drop and recreate all (standard) indices on the database. This has little use except in the case where indices got corrupted by disk errors or such.
- `convert`
  The entry point `convert` has a different syntax than the others.

  ```
  texis v2db=DB v2profile=PROFILE v4profile=PROFILE□↪
     ↪dowalk/convert.txt
  ```

  It is used to convert Webinator 2 profiles to Webinator 4 profiles (as well as possible). Set `v2db` to the full path to the existing Webinator 2 database containing the profile to convert. Set `v2profile` to the name of the Webinator 2 profile in the specified database to convert. Set `v4profile` to the name of the new Webinator 4 profile to create in the global database.

  A walk is NOT started. After conversion you would select the new profile, make any adjustments or fixups, then start a new walk.

4.7.2 Using gw

Another way to to run Webinator from the command line is with the Webinator 2 gw program which is bundled with Webinator 4. When used "normally" as it always has been with Webinator 2 it will produce Webinator 2 compatible databases. These databases are not compatible with Webinator 4's `dowalk` or `search` scripts. They are compatible with the old Webinator 2 search scripts you may have.

The new `texis` program in Webinator 4 may be used with your Webinator 2 scripts and databases. You may need to remove old locks from the old databases first though. Do this by deleting the `SYSLOCKS` file from the database directory. On Linux systems you may need to remove the shared memory segment for the database with the `ipcrm shm SHMID` command. List shared memory segments with `ipcs -m`.

`gw` may also be used, in a limited capacity, as a driver for `dowalk`. When you supply the `-gw4` option to gw it will create Webinator 4 profiles and run `dowalk` on them. By default it will create a profile named

`lastrun-gw4`. You may specify what profile to create by using the `-save=PROFILENAME` option. In gw4 more the argument to `-d` will be taken as the name of the `dataspace` directory rather that the `database` directory. `gw -gw4` may be used to perform the following actions:

Create a new empty database:

```
gw -gw4 -d/htdocs/webinator/newdb
```

Create a new database and perform a walk to it:

```
gw -gw4 -d/htdocs/webinator/newdb http://www.mysite.com/
```

Add single pages to an existing database:

```
gw -gw4 -d/htdocs/webinator/newdb -g□↪
   ↪http://www.mysite.com/somepage.html□↪
   ↪http://www.mysite.com/somedir/anotherpage.html
```

Gw4 mode does its work by saving a profile in Webinator 2 style, then calling dowalk's import routine to convert those settings to Webinator 4 style. It then calls dowalk again to either walk the specified site(s) or fetch the specified page(s). With a verbosity of 4 (`-v4`) or higher gw will print out the dowalk command lines that it executes.

None of the database management commands such as `-rewalk`, `-wipe`, etc. are supported. Most of gw's other options work as expected. See "`Profiles`" (4.3.13) for notes about profile importing.

4.8 Running the Search Interface

See section 5.1.

Chapter 5

Procedures and Examples

5.1 Searching your Index

Search the pages you have indexed by entering the following URL into your favorite Web browser:

```
http://www.mysite.com/cgi-bin/texis/webinator/search/
```

or for Windows:

```
http://www.mysite.com/scripts/texis.exe/webinator/search/
```

The above is a virtual path comprised of 2 parts. "`.../cgi-bin/texis`" is the Texis Web Script interpreter and "`/webinator/search`" is the path to the search script relative to the web server's document root.

You may have to use a slightly different URL if you specified a different CGI directory during installation.

The URL given above will search the live database specified in the default profile called "`default`". If that profile is not found it will try to search the default walk database, `INSTALLDIR/texis/db` on unix or `INSTALLDIR\texis\db` on Windows.

You may specify an alternate profile by including its name in the URL.

```
.../webinator/search/?pr=MYPROFILE
```

Where `MYPROFILE` is the name of the profile you wish to use. The search will use the live database specified by that profile.

You may also specify a database to search instead of a profile.

```
.../webinator/search/?db=DATABASE
```

Where DATABASE is the name of the database you wish to use. This would generally be the live database for a given profile which may be found as the first item listed on the administrative interface's Walk Settings page. Databases used this way must exist under the texis subdirectory of the installation directory. What you specify for DATABASE is only the portion of the path and name under the texis directory. For example, to search the database /usr/local/morph3/texis/myprofile/db2 you would use:

```
.../webinator/search/?db=myprofile/db2
```

When using a database instead of a profile the look and feel settings will be those that were live when the walk of that database was performed. The profile will not be consulted for more recent changes. A benefit of not consulting the profile, however, is some increased search speed, which may be useful on a very heavily searched system. A disadvantage of specifying the database is that it will no longer be correct if a new walk is performed.

To get help on constructing queries click on the Advanced button of the search form. On the advanced search form you will find hyperlinks into the search help, which is also included in this manual in section 7.

To place the search form onto your existing web page(s) call up the Live Search from the administrative interface main menu (or the URL you determined from the above). This will bring up the search form. Use your web browser's view page source option (MSIE: TopMenu->View->Source, Netscape: TopMenu->View->Page Source) to get the source of the page. Cut everything between and including the <FORM and </FORM>. That form may then be pasted into the web page(s) of your choice. You may also rearrange the look of the form as long as the variables are still present. If you have categories there will be a cq select list in the form. You may leave this out if you always want to search everything. Or you may make it a hidden variable with a fixed value if you always want to search the same section.

5.2 Similarity Searching

The search script has a feature called "Find Similar" which allows a user to click on a search result record to find more pages within the database similar to that one. This feature may also be accessed from any web page by placing the appropriate URL on it. You may search for pages in your database that are similar to any other web page whether it's in the database or not. The URL for finding similar pages has the form shown below.

Note: On Windows the /cgi-bin/texis/ *portion of the following URLs will be something like* /scripts/texis.exe/ *but may vary depending upon your installation.*

```
http://www.mysite.com/cgi-bin/texis/webinator/search/similar.html?↪
   ↪pr=default&ref=http://somesite/somepage.html
```

If the page containing the similarity URL resides on the same server as the search the http://www.mysite.com portion may be omitted:

```
/cgi-bin/texis/webinator/search/similar.html?↪
   ↪pr=default&ref=http://somesite/somepage.html
```

If the profile to be searched is "default" the pr=default& portion may be omitted:

```
/cgi-bin/texis/webinator/search/similar.html?↪
   ↪ref=http://somesite/somepage.html
```

If the profile to be searched is anything other than "default" that must be specified instead of `default`:

```
/cgi-bin/texis/webinator/search/similar.html?↪
   ↪pr=myprofile&ref=http://somesite/somepage.html
```

If the page to be located is the page the URL is on the `ref=URL` portion may be omitted:

```
/cgi-bin/texis/webinator/search/similar.html
or
/cgi-bin/texis/webinator/search/similar.html?pr=myprofile
```

The similar function will lookup the desired URL in the database or, if it's not in the database, fetch it from the webserver. It will then search the database looking for indexed pages similar to the specified page.

You could place a URL like this on all of your pages so users could, with one click, find all pages on your site similar in content to the one they were reading.

5.3 Page Exclusion, Robots.txt, and Meta-robots

On the first access to a site the file `/robots.txt` will be retrieved, if its exists. Settings there will be respected. Any encountered URL that is disallowed by `robots.txt` will be discarded. Meta robots is also respected for each page retrieved. See `http://www.robotstxt.org/wc/exclusion.html` for the robots.txt and meta robots standards.

If there are any HTML trees that you don't want indexed you may want to setup a `robots.txt` file, meta robots within the HTML pages, or use the various exclusion options to Webinator. For example: if you had a "text only" version of your web server that duplicated the content of your normal server you would not want to index it. (On the other hand if most of your meaningful text is contained in graphics, Java, or JavaScript you may want to walk the text tree instead of the normal one, since graphics and Java are not searchable.)

Suppose your "text only" pages were all under a directory called `/text`. The simplest way to prevent traversal of that tree would be to use the exclusion or exclusion prefix.

The exclusion would look something like this:

```
/text/
```

The exclusion prefix would look something like this:

```
http://www.mysite.com/text/
```

That will prevent retrieval of any pages under the `/text` tree. This does not prevent other Web robots from retrieving the `/text` tree. To setup a permanent global exclusion list you need to create a file called

`robots.txt` in your document root directory. The format of that file is as follows:

```
User-agent: *
Disallow: /text
```

Where `*` is the name of the robot to block. `*` means any robot not specifically named (all robots in this case since no others are named). Or you could specify the name of the robot. For Webinator it would be `Webinator`. You may specify several "Disallow"s for any given robot (see below). The "Disallow"s are simple path prefixes. They may not contain wildcards.

You may also specify different "Disallow" sets for different robots. Simply insert a blank line and add another "User-agent" line followed by its "Disallow" lines.

Here's a larger example:

```
User-agent: *
Disallow: /text
Disallow: /junk

User-agent: Webinator
Disallow: /text
Disallow: /webinator

User-agent: Scooter
Disallow: /text
Disallow: /junk
Disallow: /big
```

The `Scooter` robot will be blocked from accessing any pages under the `/text`, `/junk`, and `/big` trees. `Webinator` will be blocked from accessing any pages under `/text` and `/webinator`. All other robots will be blocked from accessing pages under `/text` and `/junk`.

Use of `robots.txt` is not enforced in any way. Robots may or may not use it. Webinator will, by default, always look for it and use it if present. This may be disabled by turning off "Respect robots.txt". When using robots.txt you may still use "Exclusions" for manual exclusion.

Meta robots provides another method of controlling robots such as Webinator. Any HTML may contain a meta tag in the source of the form

```
<meta name="robots" content="WHAT-TO-DO">
```

`WHAT-TO-DO` may contain any of the following keywords. Multiple keywords may be used by placing a comma(`,`) between them.

Like `robots.txt` this is not enforced in any way. Robots may or may not use it. Webinator always indexes and follows hyperlinks by default so it only looks for `NOINDEX` and/or `NOFOLLOW` and/or `NONE`.

Table 5.1: Meta-Robots Flags

Keyword	Meaning
INDEX	Index the text of this page
NOINDEX	Don't index the text of this page
FOLLOW	Follow hyperlinks on this page
NOFOLLOW	Don't follow hyperlinks on this page
ALL	Synonym for INDEX,FOLLOW
NONE	Synonym for NOINDEX,NOFOLLOW

5.4 Indexing Other Sites

You may index a site other than your own by specifying its URL just as you would for your own site.

```
http://www.anothersite.com
```

Please be kind when indexing other sites. Many are low bandwidth or heavily used already and won't appreciate being hit hard. If you want to index any significant number of sites, please contact Thunderstone, as we may have what you want already. Remember that we are one SQL statement away from turning off any individual free Webinator license.

5.5 Indexing Individual Pages

To add an individual HTML page to the database, but not go after any of its references, add it to the `Single Page` list box.

5.6 Reindexing on a Schedule

It is often desirable to reindex a given site on a regular basis because of continuously changing content. You may specify a `Rewalk Schedule` to handle this for you.

It is also useful to perform a single rewalk at a later time or date to avoid overloading a web server during heavy use periods.

5.7 Checking for WEB Server Errors

When you start a walk you will be sent to the walk status page. You may also reach that page at any time by selecting `Walk Status` from the menu. This page will show you the summary status of the running walk. When the walk completes you will see a summary of the walk as well as a list of any errors encountered. Following the error list is a list of duplicate pages encountered.

You may also view document linkage and info and errors from the `List/Edit URLs` page (4.3.5) from the menu.

5.8 Removing Pages from the Database

Use the `List/Edit URLs` menu (4.3.5) to find and delete specific URLs from the the database. You may delete individual pages or many pages at once using wildcards.

5.9 Erasing the Entire Database

If you decide to wipe out your existing database and it's settings to start over go to "Profiles" and click "Delete" next to the profile you wish to delete. This will completely remove the selected walk database and all options related to it.

5.10 Using Multiple Databases

Once you have a live searchable database you may want to build a separate one to contain different kinds of pages or to experiment with, without destroying your live database. Use the `Profiles` menu to create a new profile and database. You create the new profile with default settings or with a copy of the settings from another profile.

Chapter 6

Reference

6.1 Database and File Usage

`Webinator` maintains a database that contains text from HTML pages, links to other pages, and a list of categories.

When the `Webinator` walker runs it creates a new database, under your specified data directory, to hold the new walk. It then dispatches a separate process for each web site it needs to visit and another to handle all of the "Single Pages". Each of these retrieves all of the pages in it's base list and stores the text of the HTML page to the html table and the hyperlinks to the refs table. All of the desirable URLs from the page that have not been seen before are placed into an internal "todo" list. After all of the base URLs are processed the process repeats with the internal todo list. When there's nothing left in the todo list processing is complete.

Once all of the walking is complete the indices needed for searching are created on the data. Then the new database is flagged as the "live" one and the old database is deleted. Therefore your disk must have sufficient space for 2 complete databases plus temporary space used during the indexing step.

The databases are stored under your specified data directory. The databases are called `db1` and `db2`. Webinator alternates between using these two names.

Note that the above applies to a walk type of `New`. During a walk type of `Refresh` only one database, the "live" one, is used.

Webinator also maintains a file containing the detailed report for each walk. This file has the same name as the database with `.long` appended to the end. Also, a single file called `summary` is maintained with short summary information about the state of the database.

Given a data directory named `.../default` there may also be the following:

`.../default/db1` an actual walk database

`.../default/db2` an actual walk database

`.../default/db1.long` detailed walk report. Displayed when viewing `Walk Status`

`.../default/db2.long` detailed walk report. Displayed when viewing `Walk Status`

`.../default/summary` summary walk report. Displayed as `Walk summary` when viewing `Walk Settings`

Webinator, being based on Texis, also has the notion of a global "default" database. This database resides in the installation directory. On unix it is called `INSTALLDIR/texis/testdb`. On Windows it is called `INSTALLDIR\texis\testdb`. This database is used to hold all of the profile and account settings. It does not contain any walked data. It is recommended that you *not* use this as your data directory.

Each setting has a record in the `options` table of the default database. See section 6.2 for the list of fields in the table. At each complete rewalk the current options settings are copied into an options table in the walk database. These options are not changed as settings are modified and are not otherwise used unless a search is performed setting the database with `db` instead of setting the profile with `pr`.

6.2 Database Tables and Fields

These are the walk database fields:

Table 6.1: html table

Field	Description
id	Unique record id
Hash	Document hash for duplicate content detection
Size	Size of retrieved html document
Visited	The date the page was modified (or fetched if modified not set)
Dlsecs	The number of seconds to fetch the page
Depth	The number of URLs traversed to reach the page
Url	The URL of the real HTML page
Title	The Title of the page
Body	The textual content of the page
Keywords	The `keywords` meta data from the page
Description	The `description` meta data from the page
Meta	Other meta data from the page, separated by newlines
Catno	List of categories to which the URL belongs

Table 6.2: refs table

Field	Description
Url	The URL of the HTML page
Ref	The URL of a reference (link) on the HTML page
id	Unique record id.

These are the options table fields (maintained in the default database):

You can look at the SYSCOLUMNS and SYSINDEX tables of the database for details about the field types, sizes, and indices.

Table 6.3: categories table

Field	Description
Catno	The number for the category
Url	The URL pattern for the category
Category	The name of the category

Table 6.4: error table

Field	Description
Url	The URL of the an HTML page that could not be retrieved
Reason	The reason it could not be retrieved
id	Unique record id (includes timestamp info).

Table 6.5: querylog table - (only used if query logging is enabled)

Field	Description
id	Contains the date and time of the query (unique record id)
Client	The hostname of the web client that performed the query
Query	The user's query as entered

Table 6.6: options table

Field	Description
id	Unique id for the record
Profile	The name of the profile that the record belongs to
Name	The name of the setting
Type	The data type of the setting (always String)
String	The value of the setting
Int	Unused
Float	Unused
Strlist	Unused

6.3 Customizing the Search

You may make common changes to Webinator's search appearance by using `Search Settings` from the administrative interface main menu. But you are not limited to those features. You may change any and all aspects of the search program's appearance and behavior by modifying the supplied `search` script or writing an altogether new one.

For details on programming with Texis Web Script (Vortex), see the manual at the Thunderstone web site, `http://www.thunderstone.com`.

The following describes some important points about the internals of the default search script that comes with Webinator. The search script is fairly heavily commented to aid in finding your way around within it.

The `init` function is called from every entry point. It is a good place to place settings that should always or most times apply. It understands the old style specification of database by the `db` variable as well as the new method of extracting the database name from the profile named by the `pr` variable.

The `top` function displays the common HTML for the beginning of every page generated by the script. This does not include the search form. It is where you would place styles and navigation menus.

The `bottom` function is the complement to the `top` function. It displays the common HTML footer for the end of every page.

The `showform` function displays the search form with all current settings indicated.

The `qpar` and `fpar` functions process the user's form submission and apply appropriate search parameter settings.

The `credit` function displays the Thunderstone credit on the search results. This is required for free users but may be changed or emptied for paid users.

The `result` function is called for each matching record to display. It then calls the configured `result*` function to generate the desired output style.

The `mlt` function is called to setup the search when the end user selects "`Find Similar`" (aka More Like This).

The `similar` function may be called directly to find pages within the database that are similar to the content of the URL specified. It has the same concept of "`Find Similar`" but will work on any specified URL, not just those displayed as the result of a search. It would be invoked something like this on any HTML page.

```
<a href="/cgi-bin/texis/webinator/search/similar.html?↪
   ↪pr=default&ref=http://somesite/somepage.html">↪
   ↪Find pages similar to somepage.html</a>
```

or

```
<a href="/scripts/texis.exe/webinator/search/similar.html?↪
   ↪pr=default&ref=http://somesite/somepage.html">↪
   ↪Find pages similar to somepage.html</a>
```

Set "`default`" above to the search profile you're using.

It will lookup that URL in the database or, if it's not in the database, fetch it from the webserver. It will then search the database looking for indexed pages similar to the specified page.

The `main` function is the standard Vortex default entry point. This is the function that is first called when users click "Submit" on the search form.

The `search` function does the core work of finding matching documents within the database. It calls `showform` and `qpar` then starts searching. For every match the `result` function is called. The `summary` function is called before the first match is displayed to display the search results summary. It is called again at the end of the results list.

The `putmsg` function handles errors that may occur and displays them in a somewhat more user friendly fashion. See the vortex manual for details about how `putmsg` is used to capture errors.

6.4 Customizing the Walker

You may make many changes to Webinator's walk behavior by using `Walk Settings` from the administrative interface main menu. But you are not limited to these features. You may change any and all aspects of the walker's behavior by modifying the supplied `dowalk` and/or `webinatoradmin` script.

For details on programming with Texis Web Script (Vortex), see the manual at the Thunderstone web site, `http://www.thunderstone.com`.

The following describes some important points about the internals of the dowalk script that comes with Webinator. The dowalk script is fairly heavily commented to aid in finding your way around within it.

The dowalk script actually consists of 2 vortex script files. `dowalk` contains the walker/indexer and settings reading code. It includes the second file which is a vortex source module called `webinatoradmin` and must be in the same directory as dowalk. The webinatoradmin module provides the management interface that is used from a web browser.

Dowalk is not compatible with old-style gw databases. It can, however, be made compatible. There are comments throughout the script containing the word "COMPATIBILITY" that indicate where and what kind of changes to make. The most significant differences are the addition of several fields to the html table and the keeping of the leading `http://` on URLs in the database.

The `dispatch` function is the primary external entry point for performing a new walk. It load settings, sets up logging and databases, then invokes other processes in parallel (according to maximum servers setting). When all of the walking is complete it removes commonality from pages (if that option is set), creates the indices needed for searching the database, then makes the new database live and deletes the old database.

The `refresh` function is the entry point for a `refresh` walk (as opposed to `new`). It sets up logging and such then called `dorefresh` to do the work. `dorefresh` loops over all URLs in the database and queries the webserver if there is a newer version to download. Any new hyperlinks found on downloaded pages are also downloaded and added to the database. Any pages that return error will be deleted from the database.

The `stop` function is an external entry point that is used to signal (using `<loguser>`) a walk that is in progress that it should stop. The walkers check for this signal (using `<userstats>`) at various points and will quit when it is detected.

The `reindex` function is an external entry point that is used to drop and recreate the Metamorph index on the html table. This is needed after changing the word definition expressions.

The `remakeindex` function is an external entry point that is used to drop and recreate all indices on the database. It it only for use if one or more non-Metamorph indices get corrupted by disk errors or such.

The `recat` function is an external entry point that is used to recategorize the html table based on the current (presumably changed) categories.

The `ifmodified` function is an external entry point that is used to tell the dispatcher to run only if chkneedwalk indicates a walk is needed.

The `usage` function is called when you invoke dowalk incorrectly and prints a terse summary or correct usage options.

The `doplugin` function handles files that are not HTML or text, such as PDF and MSWord. It determines the correct options for `anytotx` based on the fetched page's mime type or extension. It then calls the `dofilt` function which actually runs `anytotx` to perform the conversion to text and the extraction of meta information such as Title. It will make up a title for the document if none is returned by `anytotx`.

The `settings` function calls the defaults, readsettings, and applysettings functions, in order. This function is called by most entry points to get default and current settings for a given profile before proceeding with any work.

The `updatemmindex` function is called (sometime after having called settings) to create or update the Metamorph index on the html table.

The `maketables` function is called (sometime after having called settings) to create all of the Webinator tables. This function does nothing for Webinator-only licenses. For Webinator-only licenses the tables are created automatically by Texis when the database is created. The schema may not be changed. If you want to modify dowalk to work with gw style databases, you will need to create the database and tables with gw before running dowalk.

The `walk` function is the core which walks all desired URLs on a single site. It always processes breadth first (ie it gets all URLs at a given depth before proceeding to the next level down). Any desired URLs that reside on a different site are placed into the database's todo table for processing by the dispatcher.

The `fetchset` function is used in various places to fetch one or more URLs (using the maximum threads setting) simultaneously.

The `manglepage` function is called before extracting text and hyperlinks from an HTML page. It allows the page to be modified before processing. This is where the ignore/keep tags are handled.

The `getrobotstxt` function fetches the robots.txt file from a given site and checks for any exclusions for webinator. These exclusions are later added to the list of url rejection patterns.

The `chkneedwalk` function is called to check if a rewalk is required. It fetches the page to see if the modification date has changed. Or, if the web server does not provide a modification date it compares the content to what it was previously. It sets an internal flag if a rewalk is needed.

The `putmsg` function intercepts error messages to provide special handling for some, and recording of most.

The `go` function is an external entry point used by the dispatcher when it starts up child processes to walk a specific site or set of URLs.

The `singles` function is an external entry point that is used to fetch all of the single page URL. It is called by the dispatcher as the first parallel process. Therefore single pages will generally be fetched earliest in a new walk.

The `rmlocks` function is used to remove any stale locks and monitor processes on a database and dismantle the locking structure. This is done before physically removing a database from the system.

The `geturl` function is a utility function that may be used to find out what the walker will think about a given URL using the current walk settings. It is invoked as follows:

```
texis profile=PROFILE top=THEURL dowalk/geturl.txt
```

This can generate a lot of output for a page of any size so you may want redirect it to a file that you can examine with your favorite viewer/editor.

```
texis profile=PROFILE top=THEURL dowalk/geturl.txt >SOMEFILE.txt
```

The `getrobots` function is a utility function that may be used to find out what the walker will think about a given robots.txt using the current walk settings. It is invoked as follows:

```
texis profile=PROFILE top=THEURL dowalk/getrobots.txt
```

This can generate a lot of output for a page of any size so you may want redirect it to a file that you can examine with your favorite viewer/editor.

```
texis profile=PROFILE top=THEURL dowalk/getrobots.txt >SOMEFILE.txt
```

6.5 Third-Party Software

Webinator may contain and utilize the following third-party software to enhance its functionality, depending on the version purchased. Note that your usage and rights to such third-party software are governed by the appropriate licenses originating with that software, not your License Agreement with Thunderstone - EPI.

- **SpiderMonkey (JavaScript-C) Engine**
  The Mozilla Project's SpiderMonkey (JavaScript-C) engine may be used for walking JavaScript links. See the `txjs.tar` or `txjs.zip` file in the `texis` dir of the Webinator installation directory for more information, including the license. Complete documentation and source code to the SpiderMonkey engine is available at:
  `http://www.mozilla.org/js/spidermonkey/`

Chapter 7

Search Interface Help

7.1 Forming a Query

The Webinator's search can be as simple or as complex as you need it to be. Usually you will just need to enter a few words that best describe that which you are trying to locate. To perform more complicated searches you might use any combination of logic operators, special pattern matchers, concept expansion, or proximity operations.

Example: `nature conservation organization`

7.1.1 Query Rules of Thumb

- If you get too many junk or nonsense answers, try:
  - Add some more words to your query.
  - Decrease the range of the `Proximity` control.
  - Change the `Word Forms` control to `Exact`.
  - Look at the Match Info and see why they are showing up.
  - Use the Exclusion Operator `(-)` to remove unwanted terms.
  - If you are searching for a phrase, hyphenate the words together.
- If you don't get any answers, or just too few:
  - Remove some more words to your query.
  - Examine your spelling.
  - Increase the scope of the `Proximity` control.
  - It just might not be there?

7.1.2 Overview of Query Abilities

The Webinator is based on Texis and as such it shares its text query abilities with all of Thunderstone's products. Throughout our documentation you will see references to Metamorph or Texis. This is because all of our products share a common text query language. This document provides only a brief overview of this language.

If you'd like to know more see the online manual at
`http://www.thunderstone.com/site/texisman/link_mmq.html`.

7.1.3 Controlling Proximity

Mastering the usage of proximity gives the ability to locate answers with greater precision. The Webinator input form gives you several options to control the search proximity:

line All query terms must occur on the same line

sentence Query items should all reside within the same sentence

paragraph Within the same paragraph or text block

page All items must occur within same HTML document (the default)

A bar-graph display will be shown any time a ranking search was performed (eg. all searches except `Show Parents`).

7.1.4 Ranking Factors

The ranking algorithm takes into consideration relative word ordering, word proximity, database frequency, document frequency, and position in text. The relative importance of these factors in computing the quality of a hit can be altered under `RANKING FACTORS` on the `Options` page.

7.1.5 Keywords Phrases and Wild-cards

To locate words, just type them in as you would in a word processor. Letter cases will be ignored.

The wild-card character `*` (asterisk) may be used to match just the prefix of a word or to ignore the middle of something.

If the item you wish to locate is more complicated than the simple `*` wild-card can accomplish, try using the regular expression matcher (http://www.thunderstone.com/texis/site/pages/regexp.html).

To locate a number of adjacent words in a specific order, surround them with `"` (double quotation) characters. Putting a `-` (hyphen) between words will also force order and one word proximity.

* see `Word Forms` (7.2)

Table 7.1: Query examples

Query	Locates
`john`	john, John
`"john public"`	John Public
`web-browser`	Web browser, web-browser
`John*Public`	John Q. Public, John Public
`456*a*def`	1-456-789-ABCDEF
`activate`	activate, activation, activated, ... *

7.1.6 Applying Search Logic

Texis and Metamorph use set logic for text queries. Set logic is easier to use and provides more abilities than boolean. The examples below make reference to single keywords, but keep in mind that each keyword can represent an entire list of things or any of the special pattern matchers.

Sets (or lists) of things are specified by placing the elements within parenthesis, separated by commas. Example: *(bob,joe,sam,sue)* . In the examples below, you could replace any of the keywords with a list like this.

The default behavior of the search is to locate an intersection (or 'AND') of every element within a query. This means that the query: *"microsoft bob interface"* is the equivalent to the boolean query: *"microsoft AND bob AND interface"* .

- (without) The `-` (minus) is the most commonly used logic symbol. It means the answer should EXCLUDE references to that item.

+ (mandatory) The `+` (plus) symbol in front of a search item means that the answer MUST INCLUDE that item. This is generally used in conjunction with the permutation operation.

@N (permute) The `@` followed by a number indicates how many intersections to locate of the terms in your query. This may be confusing at first, but it is very powerful.

Table 7.2: Search Logic Examples

Query	Finds
`bob sam joe`	Bob with Sam and Joe
`bob sam -joe`	Bob with Sam without Joe
`bob sam joe @1`	Bob with Sam, or Bob with Joe, or Joe with Sam
`A B C D @1`	AB or AC or AD or BC or BD or CD
`+A B C D @1`	ABC or ABD or ACD
`A B C -D @1`	(AB or AC or BC) without D

The plus(+) and minus(-) operators must be attached to the term to which they apply. There must be a space between the operator and any preceding term.

Correct	Incorrect
`bob +sam -joe`	`bob + sam - joe` `bob+sam-joe`

7.1.7 Natural Language Query

You may enter a query in the form of a sentence or question. The software will automatically identify the important words and phrases within your query and remove the "noise words".

Example: What is the state of the art in text retrieval?

The software will search for: `state of the art` AND `text` AND `retrieval`

7.1.8 Using the Special Pattern Matchers

These pattern matchers are used to locate hard-to-find items within text:

- Regular expression matching for complex patterns (`http://www.thunderstone.com/texis/site/pages/regexp.html`)
- Approximate pattern matching for fuzzy searches (`http://www.thunderstone.com/texis/site/pages/xpm.html`)
- Numeric pattern matching for finding quantities (`http://www.thunderstone.com/texis/site/pages/npm.html`)

If improperly used these pattern matchers can slow queries. Therefore they require other keyword(s) in the query and are disabled entirely under Page proximity. For more details see the Vortex manual on Query Protection (`http://www.thunderstone.com/site/vortexman/link_qprot.html`).

Table 7.3: Pattern Matcher Examples

Query	Matcher	Finds
`ronald %regan`	Approx	Ronald Raygun, Ronald Re-an, Ronald 8eagan
`%75MYPARTNO9045d/6a`	Approx	Anything within 75% of looking like MYPARTNO9045d/6a
`/19[789][0-9]`	RegEXpr	1970-1999
`/[1-9]{3}\-=[0-9]{4}`	RegEXpr	Phone numbers: 555-1212, 631-8544
`#87`	Numeric	four score and seven, 87
`#>0<1`	Numeric	Fractions like 9/16, 55%, 0.123, 15 nanoseconds

Table 7.4: Word Form Examples

Word	president
EXACT	president
PLURAL	(above) + presidents president's
ANY	(above) + presidential presidency preside presides presiding presided
Word	tight
EXACT	tight
PLURAL	(above) + tights
ANY	(above) + tightly tightening tightened tighter tightest
Word	program
EXACT	program
PLURAL	(above) + programs program's
ANY	(above) + programming programmatic programmed programmer programmable

7.1.9 Invoking Thesaurus Expansion

Metamorph and Texis have an edit-able vocabulary of over 250,000 word and phrase associations. Each entry is generally classifiable by either its meaning or part of speech.

To expand the meaning of a word or phrase within your query, precede it with a ~ (tilde) character.

7.2 Using Word Forms

The `Word forms` options give you control over how many variations of your query terms will be sought in your search.

Exact: Only exact matches will be allowed. (the default)

Plural & possessives: Plural and possessive forms will be found. (s, es, 's)

Any word forms: As many word forms as can be derived will be located.

We call this morpheme processing, and it is generally smarter than a traditional "stemming" algorithm. It doesn't just rip the end off a word, it actually checks to see if it could be a valid form of the search term. More information is available at
`http://www.thunderstone.com/site/texisman/link_ling.html`.

Notes: Thesaurus terms are also treated in the same manner. Words smaller than 4-5 characters will not be morpheme processed.

7.3 Controlling Proximity

These options give you control over the region in which a match must be found.

line: match terms must be located within the same line.

sentence: all terms within the same sentence.

paragraph: match terms must be located within the same paragraph.

page: (default) all terms within the same document.

In all cases the best possible matches for your query are located and ordered by decreasing quality. A bar graph is produced to indicate the quality of each answer.

7.4 Interpreting Search Results

Note: *The look and feel described here is for the standard search interface. The interface may have been customized by the web site administrator.*

When a query is submitted it will come back with another query form and up to 10 matching documents. If there are more than 10 answers, a link at the top and bottom of the list will allow you to view the next 10 in sequence.

The input form at the top allows you to further tailor your query to home-in on the desired answers, or to submit a completely new query without having to navigate back to the original input form.

Each answer in the result set will have a format similar to the following:

```
1: THE DOCUMENT TITLE (hyperlink to original)    84%*******___
   This is the document abstract. It consists    Size: 11K
   of the text around the first hit within the   Depth: 3
   matching document...                          Find Similar
   http://www.thesite.com/thepage.html           Match Info
                                                 Show Parents
```

The components of each result are:

- Result number
- Document title *(Clicking on this will take you to the original document)*
- Abstract *(The first few hundred characters of the document)*
- Match quality graph. `84%*******___` *(Only shown if relevance ranking was used)*
- Size *(How big is the original document)*
- Depth *(How many clicks from the top of the site)*
- Find Similar *(Find other documents similar to this one)*
- Match Info *(View the matches and other information about the document)*
- Show Parents *(List pages that link to this one)*

7.4.1 Viewing Match Info

The `Match Info` link will show you the context of your answers within the matching document. Matching words will be shown as hyperlinks. Clicking on any match term will take you to the next matching term. A summary at the top of the in-context view shows information about the document, including the time it was last modified.

7.4.2 Finding Similar Documents

The `Find Similar` link will find documents that are similar to the corresponding result. It does this by reading the original document to ascertain its main subject matter, and then conducting a relevance ranked search for those subjects.

Result documents are ordered from best to worst match. The bar graph display will indicate the overall quality of the match.

Note: The document you click on may not be ranked as the best match. This is because other documents may contain more information about the overall subject matter than the original.

7.4.3 Showing Document Parents

Often times it is difficult to navigate using a search engine because there is no *back-link* present on the matching document. The `Show Parents` link solves this.

This link will show other documents that contain hyperlinks to the one you click on. In other words, it is an automated back button.