ht://Check

Gabriele Bartolini <angusgb@users.sourceforge.net>, Comune di Prato version 1.2.4, July 4th, 2006

User guide of ht://Check program

Contents

1 Foreword by the author

Dear ht://Check user,

If you read this, it means that at least you downloaded this extremely useful tool for Webmasters and, more in general, for Web site maintainers.

I hope you could start using this tool and get into it in a short time; and also you could use it on a daily basis, for daily operations of Web sites administration, management and control of documents.

Just know that where I work, in Prato, Italy, we offer citizens, services regarding our government institution (city council) and, technically speaking, we have to manage more than 35 thousand HTML documents made by various Web publishers; but we have to control and to guarantee the site integrity (i.e. there are no broken links).

And our documents reside on more than one Web server. I remember that the very very first version of ht://Check was started by me in 96, as my first big project made in C; it didn't use any HTTP call at all (I hardly knew what HTTP was after finishing high school!), just local calls. I called it **htmlcheck**.

I was so happy to see it working under Linux, with use of memory allocation structures such as lists, queues, binary trees and so on ... it could manage thousand documents, but only if they were on the same machine. You can imagine how sad I became when we started to move documents on several servers, or virtual hosts (it would not have been much of a problem to modify the code to handle them, as long as documents are all on the same host), and we began to use server-side techniques for dynamic publishing on the Web.

I had to rethink the whole thing. Fortunately I had just started to use ht://Dig as our main search engine and started to help the development of the project as contributor. I found that there were many similar aspects in the two programs, especially as far as the **spider** part is concerned.

I asked Geoff if he would have minded to see me using part of the code of ht://Dig used in a new project, a link checker. I guess it was the beginning of 1999 and as the program derived from ht://Dig, I decided to call it ht://Check.

ht://Dig was GPLed, so I was so happy to be kind of **forced** to release ht://Check under GPL as well; and every day I am more and more convinced it was a great choice, because many people all over the world write me, because they found a bug. So you happily modify the code to fix it, and your application becomes more and more robust. That's a victory for everyone, I guess.

I also feel to thank everybody at my workplace who let me have the time to think, design, develop and maintain this wonderful project; the ht://Dig group, in particular Geoff, Gilles and Loic for their great support to me; ht://Check users and contributors of ideas, bug discoveries and ... whatever!

Now, the main shared part between these two project is the network library, which can now handle HTTP/1.1 with persistent connections and cookies support, and it's continously developed with the help of other contributors. Soon HTTPS will be made available too.

A quick note: the very first successful run of ht://Check was in April of 2000. Now, it is heavily used more than once a day in our working environment, managing more than 4 million records on 35 thousand documents retrieved (in 1 hour).

Finally, ht://Check is an open source project and it comes for free.

Thank you.

Sincerely yours, Gabriele Bartolini

2 Introduction

***ht://Check** is more than a link checker*. It's a *console application* written for **GNU/Linux** systems in C++ and derived from the best search engine available on the Internet for free (GNU GPL): **ht://Dig** .

However, ht://Dig is not needed in order to install and run ht://Check, which is therefore totally independent: the only relationship existing between these two applications, is that ht://Check's code is partially derived from ht://Dig.

ht://Check can retrieve information through **HTTP/1.1** and store the information in a **MySQL database**, and it is particularly suitable for small Internet domains or Intranet.

Its purpose is to help a Webmaster managing one or more related sites: after a "crawl", ht://Check creates a powerful **data source** made up of information based on the retrieved documents. The kind of information available to the ht://Check user includes:

- **complete source code** for HTML documents retrieved;
- **single documents attributes** such as content-type, size, last modification time, etc.;
- information regarding the **retrieval process of a resource**, like for instance whether the resource was succesfully retrieved, or not, showing the various results (the so-called **HTTP status codes**, as ht://Check uses this protocol for crawling the Web);
- information regarding the **structure of a document**, basically its HTML link tags, and the relationships they issue, in a whole process view: basically, ht://Check is able to crawl a **Web domain** or set (in the algebrical meaning), and links create sort of **inter-documents relationships** in it. This feature, allows the user to get further information from the domain regarding:
  - **link results**: if it is either working or **broken** or redirected, or bad encoded (according to RFC1738); also at the current status, it checks whether a link is actually an anchor that does not work, or it is a javascript or an e-mail;
  - the **relationships between documents**, in terms of incoming links and outgoing ones (Web structure mining activity);
  - **accessibility checks**: from version 1.2.3, ht://Check also performs accessibility checks in accordance with the principles of the University of Toronto's Open Accessibility Checks (OAC) project, allowing users to discover site-wide barriers like images without proper alternatives, missing titles, etc.

A skinny report is given by the program htcheck, however at the current situation most of the information is given by the **PHP interface** which comes with the package and that is able to query the database built by the htcheck program in a previously made crawl. It goes without saying that you need a Web server to use it, and of course PHP with the MySQL connectivity module.

By the way, as long as after a crawl ht://Check produces a database on a MySQL server, it's needless to say that every user theoretically could build its own information retrieval interface to this database; you only need to know the structure of it, its tables and fields, and the relationships among them. Other solutions are represented by independent scripts written by using common scripting languages with MySQL connectivity modules (i.e. Perl and Python), or faster programs written in C or C++ using MySQL API or wrapper libraries (such as MySQL++ or dbconnect), or other Web driven solutions like JSP, ColdFusion. There exists an interface to ht://Check for the Roxen Internet Software (http://www.roxen.com/) written by Michael Stenitzer (stenitzer@eva.ac.at).

Something that must not be underestimated, is that ht://Check theoretically can give the user lots of information regarding the structure of a Web domain: in a few words it can be used for **Web Structure Mining** purposes.

ht://Check is distributed under the GNU General Public License (GPL). See the

ht://Check main Website is at `http://htcheck.sourceforge.net/` .

3 How it works

ht://Check is essentially a web *spider*, or *robot* or *crawler*. As well as a search engine (like ht://Dig) indexes words from the Internet, ht://Check stores HTML statements such as tags and attributes, links, URL information, and more.

At the moment, ht://Check supports only **HTTP/1.1** (and HTTP/1.0 also): future plans regard enabling the FTP, NNTP, HTTPS and also local files checks.

Everything is stored in a MySQL database, created from scratch by the application itself. You don't need to create it before, just run '`htcheck`' and every needed table will be automatically built by the program.

For information regarding the connection to the MySQL database, please consult the

3.1 The *information retrieval* module

ht://Check is made up of two logical "modules", one corcerning the information retrieval, the other one the analysis of the performed crawl.

The first step, which is the most important also, is completely performed by the '`htcheck`' program; depending on the values set in the

When `htcheck` retrieves the first document, it checks the answer that the server gave back; if the document exists (HTTP 200 **status code** is returned), and the `Content-Type` is `text/html`, `htcheck` starts parsing the document, and retrieves and stores at least all of the HTML tags and attributes that create a link (it can store all of them if you set '`store_only_links`' to false).

`htcheck` can also manage HTTP redirection (created by header "*Location*" sent by the remote HTTP server) and cookies (as defined by http://www.netscape.com/newsref/std/cookie_spec.html).

In a few words that's the main mechanism regarding the information retrieval module, but -believe me- it is not as easy as it seems! But, as far as you are concerned, I think that's enough for now.

3.2 The tables of a *ht://Check* database

First of all, you don't need to create a database for ht://Check; indeed `htcheck` will do it for you!

However, ht://Check creates a database which is made up of these tables:

- Schedule
- Url
- Server
- HtmlStatement
- HtmlAttribute
- Link
- htCheck
- Cookies (since version 1.1)
- Accessibility (since version 1.2.3)

The main task of the **Schedule** table is to manage the crawling system: by querying this table, `htcheck` knows which URLs need to be retrieved, or just checked if they exist.

The **Url** table contains info about those URLs that have been retrieved (either successfully or not): here you can find the HTTP status code returned and its reason phrase, its size, the last access time and modification time too, and more.

The **Server** table contains information about the HTTP servers that have been encountered during the crawling process.

The **HtmlStatement** table contains information about the HTML statements found in each URL; every one of them contains one and only one HTML **tag**, but can also contain one or more HTML **attributes** inside. These ones are stored in the **HtmlAttribute** table.

The **Link** table let us find and locate every link instantiated by HTML statements (or by HTTP redirections too), so we can have a referencing as well as a referenced URL, and know precisely which HTML attribute created this link.

The **Cookies** table is handled since version 1.1 and stores all the cookies that have been retrieved during the crawl and their related information.

The **htCheck** table contains general info such as start and finish time, number of connections, etcetera.

The **Accessibility** table

3.3 Getting the information stored

Our starting point is that we now have a database full of information, because `htcheck` has already finished to crawl through the web.

The very first way to get reports from a **crawl**, is to run `htcheck` with the '-s' option, which let it produce summaries (see the

The other way given by ht://Check is to use the PHP interface, which is really simple and easy to use (for installation and settings see the

As the database is now a common MySQL database, you can use whatever you want in order to to retrieve the information stored in it (Perl, C/C++ programs, JSP). You can also get them on Windows systems, just download **MyODBC**. You got lots of choices, as you can see!

4 Installation

4.1 System Requirements

In order to install and run ht://Check you need a GNU/Linux system with:

- **GNU C/C++ compiler** and **libstdc++** installed
- **MySQL** 5.x, 4.x, 3.23.x or 3.22.x
- **PHP** version 5.x, 4.x (if you want to use the interface - it should work with PHP 3 too but I can't test it anymore).

However, ht://Check compiles on other POSIX platforms: so please, if you try and successfully install it, please drop me a line with the characteristics of your system. I have used the GNU/GCC compiler in order to build it (version 2.95, 2.96, 3.0, 3.3 or 4.0.3).

The compilation process has been tested on these platforms:

- x86, Linux 2.6 (Ubuntu);
- x86, Linux 2.4 (Redhat 8.0);
- x86, Linux 2.4 (Redhat 7.3);
- x86, Linux 2.4 (Debian 2.2);
- x86, FreeBSD (4.7-STABLE);
- Alpha, Linux 2.4 (Debian 3.0);
- PPC - G4, MacOS X 10.1 SERVER Edition (statically linked);
- Sparc - Ultra60, Linux 2.4 (Debian 3.0).

4.2 Download ht://Check

You can download ht://Check from http://htcheck.sourceforge.net/ .

4.3 Decompressing the tarball

Usually you download ht://Check sources in a 'tar.gz' file. In order to decompress them with the following command:

`tar xzvf` *`filename`*`.tar.gz`

4.4 Quick Install

```
configure
make
make install
```

4.5 The configure script

For more info on the 'configure' script, run:

```
configure --help
```

4.6 Specifying the application directory

By default, ht://Check is installed into the /opt/htcheck directory. And everything is under that directory. Nothing is put out of it. If you want to specify another directory of installation, just use the configuration option –prefix=DIR. For example, if you want to install it into the /myapps/htcheck dir, just run configure with this option too:

```
configure [other options] --prefix=/myapps/htcheck
```

4.7 Specifying a MySQL directory

ht://Check needs **MySQL** support. Since newer versions of MySQL installs its various components under /usr/local, this is the default. If you have it under a different location, you can specify it with (assuming in /opt/local):

```
--with-mysql=/opt/local
```

Otherwise just use: –with-mysql or nothing.

4.8 Setting the path to libmysqlclient.so library

If you configured ht://Check with dynamic libraries linking (this is the default), you need to let `htcheck` know where they are at run-time.

In order to do this, you have two chances: as root, edit `/etc/ld.so.conf`, insert here the mysql library path (for instance /usr/local/mysql/lib/mysql) and run `ldconfig`. Otherwise, you can just put this path in the environment variable `LD_LIBRARY_PATH`, in this way:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/mysql/lib/mysql
```

Of course, if you specified a different path for MySQL, just substitute it the path above in order to make it point to the .so files for the client libraries.

4.9 Setting the path to ht://Check's man page

ht://Check comes with a simple man page, useful for reminding you the options of the application. Let's suppose you installed ht://Check in the /opt/htcheck directory, you can easily set the man application to read this page too, by adding in the user or system profile (i.e. `~/.bash_profile` or `/etc/profile`) these line:

```
export MANPATH=$MANPATH:/opt/htcheck/man
```

4.10 Installing PHP scripts

The 'php' directory contains the php scripts that you can put everywhere into the document root of your web server. By default it's installed into the php directory of the application, but you can set it to a different path with the configure option --with-php-dir=DIR.

Since PHP 4.2.x, the register_globals variable has been turned off, for security reasons by PHP members; the PHP interface, since version 1.2.1, has been modified in order to automatically be rendered 'register_globals off' compliant.

Since version **1.1** you no longer have to set asp_tags on in order to make the scripts work; also, you needn't set the handler for **.inc** files as long as there are no more files with that extension.

After this, your environment is ready. Now, you have to change the settings into the 'include/global.inc.php' file regarding the hostname, and the authentication credentials for MySQL connections. Of course you need to have the access granted to the MySQL database server from the computer you run these scripts.

From version 1.2.0, ht://Check's PHP interface can get along with 'tidy' (http://tidy.sourceforge.net), a very useful and diffused HTML validator. You can easily enable the use of tidy, by changing the value of the Tidy variable in the 'include/global.inc.php' file, pointing it directly to its complete path. Basically, ht://Check passes the HTML code it retrieved during the last crawl to tidy, getting back the results as warning and suggested source. How? Just by following the operations menu in the URL view of the PHP interface.

From version 1.1.0b9-klunk, it's possible to specify in the 'include/global.inc.php' file which database/s you want to query, by editing the $dblist variable. By default, this variable is empty, so the PHP script performs a query on the MySQL server in order to get a list of databases which may belong to ht://Check. By setting this to one or more values, this *passage* is skipped.

It is possible to customize the language sentences too. For now I only have 3 language files: english (en.inc.php), italian (it.inc.php) and german (de.inc.php, thanks to Michael Stenitzer). They're under the include dir of php scripts: you only have to change the '$Language' variable value into the 'global.inc.php' file or just leave the script detect it by itself, depending on the language settings of the browser.

If you feel like adding a new language, feel free to do it and please post it to me and I'll get it downloadable by everyone. Of course your name will be on it for ever ... ;-)

4.11 The MySQL user's privileges for ht://Check

In order to run the htcheck program, you must connect to the MySQL server as a valid user, with enough permissions. As long as the spider needs to create and drop databases, tables and indexes too, perform insert, update and delete operations you must grant to it these rights (by altering the 'user' table's contents of the 'mysql' database on the MySQL server). So, set to 'Y' these fields values:

- Select_priv
- Insert_priv
- Update_priv
- Delete_priv
- Create_priv
- Drop_priv

- Index_priv

However, you are suggested to give a look at the following

As far as the PHP interface is concerned, you may want to give the user specified in the 'global.inc.php' file at least the select privilege (jump to this

4.12 MySQL connection settings

In order to access a MySQL server, you have 2 choices:

- doing nothing: the access is made by the current user to localhost with no password specified.
- create or use an existing option file for MySQL. See the following

4.12.1 MySQL connection settings using the option file

We were saying that you can create or use an existing option file for MySQL, where you can specify the host to be accessed, the user, the password, the port and the socket.

By default, ht://Check looks for the `~/.my.cnf` file and if this is not found the global option file for mysql is searched (`/etc/my.cnf`). You can change the prefix ('my') with the `mysql_conf_file_prefix` configuration option. The group searched is [client] but it can be customised with `mysql_conf_group`.

For example, you can write the /.my.cnf file this way:

```
[client]
host=mysqlserver.mydomain.com
user=htcheck
password=ht12345
```

You can also specify a different `port` or `socket`. You are strongly recommended to change this file permissions to 600.

It goes without saying that in both cases you have to **grant permissions** to the user ht://Check is connecting as. See the previous

5 Getting started

In order to perform the first crawl, you just need to edit the configuration file, which resides in the configuration directory with the name '`htcheck.conf`' (you may use another file as configuration file, but you gotta run `htcheck` it with the '`-c`' option).

Just change the '`start_url`' attribute to whatever you want, for example:

```
start_url:  http://www.foo.com
```

Remember that every URL must start with the service name, that is to say '`http://`'.

Then set the '`limit_urls_to`' attribute to `$(start_url)`, in order to scan only the 'http://www.foo.com' website.

You may change many other attributes (database name included), but for now, in order to test if it works or not, that's enough.

You can finally enter the `bin` directory inside the 'htcheck' installation directory (by default `/opt/htcheck`) and run:

```
htcheck -vs
```

However, here are the available options (just run `htcheck --help`) and you will get this:

```
usage: htcheck  [-isvkhr] [-c configfile] [-D dbname] [--help] [--version]

Options:
        -v      Verbose mode (more 'v's increment verbosity)

        -s      Statistics (broken links, etc...) available

        -i      Initialize the database (drop a previous db)

        -k      Initialize the database (drop tables, keep the db)

        -c configfile
                Configuration file

        -D dbname
                Name of the database

        --help  Display this
        -h      Same as --help

        --version       Display version
        -r      Same as --version
```

Remember that `htcheck` always check if the database already exists in the MySQL server. If it does not exist, it is created from scratch. On the other hand, if `htcheck` is launched with the '-i' option, this database is initialized again (this means that a new crawl is performed), else the program just use a previous database, which is useful in order to get some reports like broken links and anchors, content-type summaries (in this case you gotta set the '-s' option).

Since version 1.2.0 it is possible not to drop a database, but keep it alive, and recreate the structure: in technical words, ht://Check tables are dropped and then recreated: this feature was proposed by Patrick Guillot (<pguillot@paanjaru.com>) and enables to use ht://Check within a database that can be used for other purposes as well.

6 The configuration file

6.1 General syntax

ht://Check uses a flexible configuration file. This configuration file is a plain ASCII text file. Each line in the file is either a comment or contains an attribute. Comment lines are blank lines or lines that start with a '#'.

6.2 Attributes

Attributes consist of a variable name and an associated value:

```
<name>:<whitespace><value><newline>
```

The `name` contains any alphanumeric character or underline (_).

The `value` can include any character except newline. It also cannot start with spaces or tabs since those are considered part of the whitespace after the colon. It is important to keep in mind that any trailing spaces or tabs will be included.

It is possible to split the `value` across several lines of the configuration file by ending each line with a backslash (\). The effect on the value is that a space is added where the line split occurs.

If ht://Check needs a particular attribute and it is not in the configuration file, it will use the default value which is defined in htcommon/defaults.cc of the source directory.

6.3 Inclusion and variable expansion

A configuration file can include another file, by using a special `name`, include. The `value` is taken as the file name of another configuration file to be read in at this point. If the given file name is not fully qualified, it is taken relative to the directory in which the current configuration file is found.

Variable expansion is permitted in the file name. Multiple include statements, and nested includes are also permitted. Example:

```
include: common.conf
```

6.4 Configuration attributes

Here you can find a brief explanation of ht://Check configuration attributes.

They've been grouped in these sections:

- setting the *spider* -

6.4.1 Setting the "spider"

`start_url`

This is the list of URLs that will be used to start a dig when there was no existing database. Note that multiple URLs can be given here.
Type: string
Default: `http://htcheck.sourceforge.net/`
Example:

```
start_url:      http://www.somewhere.org/alldata/index.html
```

`limit_urls_to`

This specifies a set of patterns that all URLs have to match against in order for them to be included in the search. Any number of strings can be specified, separated by spaces. If multiple patterns are given, at least one of the patterns has to match the URL. Matching is a case-insensitive string match on the URL to be used. The match will be performed *after* the relative references have been converted to a valid URL. This means that the URL will *always* start with `http://`. Granted, this is not the perfect way of doing this, but it is simple enough and it covers most cases.
Type: string
Default: `${start_url}`
Example:

```
limit_urls_to:  .sdsu.edu kpbs
```

`limit_normalized`

This specifies a set of patterns that all URLs have to match against in order for them to be included in the search. Unlike the limit_urls_to directive, this is done after the URL is normalized.

Type: string

Default:

Example:

```
limit_normalized: http://www.mydomain.com
```

`exclude_urls`

If a URL contains any of the space separated patterns, it will be rejected. This is used to exclude such common things such as an infinite virtual web-tree which start with cgi-bin.

Type: string

Default:

Example:

```
exclude_urls: students.html cgi-bin
```

`bad_extensions`

This is a list of extensions on URLs which are considered non-parsable. This list is used mainly to supplement the MIME-types that the HTTP server provides with documents. Some HTTP servers do not have a correct list of MIME-types and so can advertise certain documents as text while they are some binary format.

Type: string

Default:

Example:

```
bad_extensions: .foo .bar .bad
```

`bad_querystr`

This is a list of CGI query strings to be excluded from indexing. This can be used in conjunction with CGI-generated portions of a website to control which pages are indexed.

Type: string

Default:

Example:

```
bad_querystr: forum=private section=topsecret&passwd=required
```

`max_hop_count`

Instead of limiting the indexing process by URL pattern, it can also be limited by the number of hops or clicks a document is removed from the starting URL. The starting page will have hop count 0.

Type: number

Default: `999999`

Example:

```
max_hop_count: 4
```

`check_external`

If set to 'true', htcheck check if external Urls exist or not. An external Url is an Url which doesn't match limit configuration attributes. External URLs aren't parsed.

Type: boolean

Default: `true`

Example:

```
check_external: false
```

6.4.2 Setting the database info

`db_name`

Name of the MySQL database to be created or read.
Type: string
Default: `htcheck` (or as defined by the `--with-db-name` configure option)
Example:

```
db_name: test
```

`db_name_prepend`

String to be prepended to the MySQL database name specified. This allows to set a common string to identify all the database name used by ht://Check and to grant database privileges by using this string value. You can change the default value also by using the configure option: –with-db-name-prepend (default empty).
Type: string
Default: (or as defined by the `--with-db-name-prepend` configure option)
Example:

```
db_name_prepend: htcheck_
```

`mysql_conf_file_prefix`

Prefix for the MySQL configuration file to be searched. Default is 'my' and The file searched is usually `~/.my.cnf` (suggested). If it is not found the `/etc/.my.cnf` file is searched. For its syntax, look at the 'Option File' contents inside the MySQL documentation.
Type: string
Default: `my`
Example:

```
mysql_conf_file_prefix: htcheck
```

`mysql_conf_group`

Group to be searched inside the .my.cnf file of MySQL for getting the settings for the connection to the server. In other words, it's the section marked with [<group>] inside the MySQL option file (default is [client]).
Type: string
Default: `client`
Example:

```
mysql_conf_group: htcheck
```

`optimize_db`

Optimize the database tables at the end of the crawl. Disable it if the database server doesn't support it.
Type: boolean
Default: `false`
Example:

```
optimize_db: true
```

`sql_big_table_option`

Enable or disable this option that is useful when performing huge queries. Otherwise, sometimes when it's not set, the MySQL db server may return a 'table is full' error.
Type: boolean
Default: `true`
Example:

```
sql_big_table_option: false
```

`url_index_length`

This number specifies the length of the index of the Url field in the Schedule and Url tables of the database. You can set different values depending on the average length of the URLs that htcheck can find in your sites. If you don't want to set any limitation, just put a '-1' value.

This now allows the user to control the length of the index for the Url field in the Schedule and Url tables. This attribute may affect the performance of the crawls, as long as the length of a index can either slow down or speed up the spidering process.
Type: number
Default: `64`
Example:

```
url_index_length: -1
```

6.4.3 Setting HTTP connections

`user_agent`

This allows customization of the user_agent: field sent when the digger requests a file from a server.
Type: string
Default: `ht://Check`
Example:

```
user_agent: htcheck-crawler
```

`persistent_connections`

If set to true, when servers make it possible, htdig can take advantage of persistent connections, as defined by HTTP/1.1 (*RFC2616*). This permits to reduce the number of open/close operations of connections, when retrieving a document with HTTP.
Type: boolean
Default: `true`
Example:

```
persistent_connections: false
```

`head_before_get`

This option works only if we take advantage of persistent connections (see persistent_connections attribute). If set to true an HTTP/1.1 *HEAD* call is made in order to retrieve header information about a document. If the status code and the content-type returned let the document be parsable, then a following 'GET' call is made.
Type: boolean
Default: `true`
Example:

```
head_before_get: false
```

`timeout`

Specifies the time the digger will wait to complete a network read. This is just a safeguard against unforeseen things like the all too common transformation from a network to a notwork. The timeout is specified in seconds.
Type: number
Default: `30`
Example:

```
timeout: 42
```

`authorization`

This tells htcheck to send the supplied *username*:*password* with each HTTP request. The credentials will be encoded using the "Basic" authentication scheme. There must be a colon (:) between the username and password.
Type: string
Default:
Example:

```
authorization: myusername:mypassword
```

`max_retries`

This option set the maximum number of retries when retrieving a document fails (mainly for reasons of connection).
Type: number
Default: 3
Example:

```
max_retries: 6
```

`tcp_max_retries`

This option set the maximum number of attempts when a connection raises a
Type: number
Default: 1
Example:

```
tcp_max_retries: 6
```

`tcp_wait_time`

This attribute sets the wait time after a connection fails and the
Type: number
Default: 5
Example:

```
tcp_wait_time: 10
```

`http_proxy`

When this attribute is set, all HTTP document retrievals will be done using the HTTP-PROXY protocol. The URL specified in this attribute points to the host and port where the proxy server resides.
The use of a proxy server greatly improves performance of the indexing process.
Type: string
Default:
Example:

```
http_proxy: http://proxy.bigbucks.com:3128
```

`http_proxy_exclude`

When this is set, URLs matching this will not use the proxy. This is useful when you have a mixture of sites near to the digging server and far away.
Type: string
Default:
Example:

```
http_proxy_exclude: http://intranet.foo.com/
```

`http_proxy_authorization`

This tells htcheck to send the supplied *username*:*password* with each HTTP request, when using a proxy with authorization requested. The credentials will be encoded using the \"Basic\" authentication scheme. There *must* be a colon (:) between the username and password.
Type: string
Default:
Example:

```
http_proxy_authorization: myusername:mypassword
```

`accept_language`

This attribute allows to restrict the set of natural languages that are preferred as a response to an HTTP request performed by the digger. This can be done by putting one or more language tags (as defined by RFC 1766) in the preferred order, separated by spaces. By

doing this, when the server performs a content negotiation based on the 'accept-language' given by the HTTP user agent, a different content can be shown depending on the value of this attribute. If set empty, no language will be sent and the server default will be returned.
Type: string
Default:
Example:

```
accept_language:        en-us en it
```

`remove_default_doc`

Set this to the default documents in a directory used by the servers you are indexing. These document names will be stripped off of URLs when they are normalized, if one of these names appears after the final slash, to translate URLs like http://foo.com/index.html into http://foo.com/ Note that you can disable stripping of these names during normalization by setting the list to an empty string. The list should only contain names that all servers you index recognize as default documents for directory URLs, as defined by the DirectoryIndex setting in Apache's srm.conf, for example.
Type: string list
Default:
Example:

```
remove_default_doc: default.html default.htm index.html index.htm
```

`disable_cookies`

If set to 'true', htcheck will disable the HTTP cookies management.
Type: boolean
Default: `false`
Example:

```
disable_cookies: true
```

`cookies_input_file`

Set the input file to be used when importing cookies for the crawl; cookies must be specified according to Netscape's format. For more information, give a look at the example cookies file distributed with ht://Check. By default, no input file is read.
Type: string
Default:
Example:

```
cookies_input_file: /tmp/cookies.txt
```

`url_reserved_chars`

This string allows to customise the set of characters that can be considered as reserverd in a URL, avoiding their coding under the `RFC1738` standard. This string is used when checking whether a URL is well-encoded or not, issuing a '*BadEncoded*' state for the link which created it. The default value is slightly different from what the RFC says, giving more flexibility to the spider (it is suggested not to change it unless you are extremely sure of what you are doing).
Type: string
Default: `;/?:@&=+$,._%-#x~`
Example:

```
url_reserved_chars: \\;/?:@&=+\$,._%-#x~
```

6.4.4 Setting what to store

`max_doc_size`

This is the upper limit to the amount of data retrieved for documents. This is mainly used to prevent unreasonable memory consumption since each document will be read into memory by htcheck.
Type: number
Default: `100000`
Example:

```
max_doc_size: 5000000
```

`store_only_links`

If set to `false`, htcheck will store in the DB *every* tag he finds in every document it crawls. If set to `true`, htcheck stores only those Html attributes and statements that produce a link or set an anchor (identified by the pair tag: A, attribute: name).
Type: boolean
Default: `false`
Example:

```
store_only_links: true
```

`store_url_contents`

This attribute allows to store the contents of the parsed URLs. It is very *useful*, but can also be *dangerous*. You must know what you are doing, and if you enable this, your performances may slow down and your disk storage requirements can get extremely high. It is recommended to use this only for small crawls.
Type: boolean
Default: `false`
Example:

```
store_url_contents: true
```

`available_charsets`

This attribute specifies the set of possible *charsets* that htcheck recognises and stores into the database; other charsets will be marked as 'other'.
Type: string list
Default: `windows-1250 iso-8859-1 iso-8859-10 iso-8859-13 iso-8859-14 iso-8859-15 iso-8859-2 iso-8859-3 iso-8859-4 iso-8859-5 iso-8859-6 iso-8859-7 iso-8859-8 iso-8859-9 koi8-r koi8-u utf-8 windows-1251 windows-1252 windows-1253 windows-1254 windows-1255 windows-1256 windows-1257 windows-1258 windows-874`
Example:

```
available_charsets: iso-8859-1
```

6.4.5 Setting what to report

`summary_anchor_not_found`

Enable or disable the show of the summary of the HTML anchors that have not been found.
Type: boolean
Default: `true`
Example:

```
summary_anchor_not_found: false
```

6.4.6 Accessibility checks

`accessibility_checks`

Enable or disable the recognition of accessibility problems, using some of the checks proposed by the Open Accessibility Checks project by the Adaptive TechnologyResource Center

at the University Of Toronto. From version 1.2.3, ht://Checks internally stores this kind of information in the 'AccessibilityChecks' table using the code number specified in OAC (http://oac.atrc.utoronto.ca).
Type: boolean
Default: `true`
Example:

```
accessibility_checks: false
```

7 FAQ

7.1 Configuration and compilation

7.1.1 I'm compiling with gcc 3.2 and getting all sorts of warnings/errors about ostream and such

You should use the following command to configure ht://Check so it can be built with gcc 3.2:

```
CXXFLAGS=-Wno-deprecated CPPFLAGS=-Wno-deprecated ./configure
```

However, from version 1.2.2, sources have been updated in order to automatically detect the correct standard C++ library; backward compatibility C++ headers (such as fstream.h) are not used anymore in the main code, although pre-processing checks are performed for older libraries.

7.2 The MySQL database of ht://Check

7.2.1 What tables have to be created? What about the fields? and their format?

ht://Check does everything for you. It creates the database structure itself, so you don't need to create it before. You just need to grant the spider enough permissions in order to do that.

7.3 Configuring the 'spider' (`htcheck`)

7.3.1 How do I change the URLs to check without going through the PHP interface?

No. There's no way to configure the spider through PHP for now. You just have to edit the configuration file (usually 'htcheck.conf').

7.3.2 If I run htcheck at the commandline, I don't see a way to change the URLs to check. I'm guessing that the Server table in the htcheck database is what I want to modify, right?

No .. you don't need to modify the MySQL database at all. Indeed it's for getting the results only. Every database is directly created by the application (from scratch). You must edit the parameters in the htcheck.conf file. You have to set one or more starting URL with the 'start_url' attribute. Then you can limit the search to a set of URLs by setting the 'limit_urls_to', 'limit_normalized' and 'exclude_urls' options. These are the most used and important, though you can use the 'bad_extension', 'max_hop_count', 'bad_query_string'. But in most of cases you only have to set the 'limit_urls_to' parameter. For instance:

```
start_url: http://www.foo.com
limit_urls_to: $(start_url)
```

The 'limit_normalized' parameter checks for every URL after it's been normalized (in this format: service://[user:password]host:port/path).

7.4 The PHP Interface

7.4.1 Despite configuring the username, password, and host in global.inc.php, I keep getting the following when accessing http://localhost/php/index.php: `Access denied for user: '@localhost' to database 'htcheck'`. Why?

The problem is that the user you are connecting to the MySQL server through the PHP scripts (the one set in the global.inc.php file) has not enough permissions. Give a look at these sections:

8 Release notes

```
Release notes for htcheck-1.2.4 - 04 Jul 2006
- Support for MySQL 5.0 server
- Accessibility checks according to the Open Accessibility Checks
  Project (OAC) by the University Of Toronto (http://oac.atrc.utoronto.ca/).
  Supported checks:
    - OAC #69: MARQUEE element should not be used
    - OAC #71: Auto-redirect should not be used
    - OAC #72: Auto-refresh should not be used
- Fixed minor bugs including:
    - OAC #37-41: wrong Hx nesting (e.g.: h2 without h1)


Release notes for htcheck-1.2.3 - 01 Jun 2004
-- Accessibility checks according to the Open Accessibility Checks
  Project (OAC) by the University Of Toronto (http://oac.atrc.utoronto.ca/).
  Supported checks:
    - OAC #1: missing ALT
    - OAC #2: ALT is the same as the file name
    - OAC #3: ALT text is not shorter than 150 characters
    - OAC #7: ALT text can't be empty if image is used as an anchor
    - OAC #37-41: wrong Hx nesting (e.g.: h2 without h1)
    - OAC #48: document language must be identified
    - OAC #50: missing TITLE
    - OAC #51: empty TITLE
    - OAC #52: TITLE is not shorter than 150 characters
    - OAC #58: Images used in INPUT controls must have ALT text
    - OAC #59: Images used in INPUT controls must have valid ALT text
    - OAC #60: Images used in INPUT controls should have short ALT text
    - OAC #61: Image used in INPUT control - ALT text should not be
      the same as the file name
    - OAC #116: deprecated use of the B element
    - OAC #117: deprecated use of the I element
- PHP interface:
  - Added support for searching information regarding accessibility checks,
    thanks to Valentina Del Sapio (Comune di Prato)


Release notes for htcheck-1.2.2 - 13 Jan 2004
- Updated to new autotools (autoconf 2.58, automake 1.7.9, libtool 1.5)
- Standard C++ library automatic detection (removes compilation warnings)
- Database changes:
  - New fields stored:
    - URL's doctype for HTML documents (Url table)
    - HTML documents' description and keywords (Url table)
```

```
- PHP interface:
  - Added doctype field for URLs query
  - Added description and keywords fields for URLs query
- Fixed minor bugs including:
  - Correct negotiation of the accepted encodings with the HTTP server
  - Charset recognition when it is given through the Content-Type HTTP header
  - Automatic recovery mechanism when a HEAD call fails with some Web servers (bug #870467)


Release notes for htcheck-1.2.1 - 27 Apr 2003
- Cookies input file management, which allows to import cookies in
  ht://Check's jar and preload them before a crawl starts
- A link's description is now stored in the database, allowing
  to see which text has been used when issuing a link
- Also, it is possible to see which tags are included inside a link: this
  is useful, for instance, to see which images act as buttons.
- added the 'store_link_info' attribute, which allows
  to control the storing of the link descriptions and linked tags.
- added the 'available_charsets', which allows to check URLs against
  a set of predefined charsets.
- fixed a serious bug which prevented referring URL to be correctly set
- code updated for new autotools (autoconf 2.57, automake 1.6.3 and libtool 1.4.3).
- minor changes.
- Database changes:
  - New fields stored:
    - URL's Charset (Url)
    - Link's description (HtmlStatement)
    - Link's position of the tag (HtmlStatement)
- PHP interface:
  - Automatically works with 'register_globals' off
  - Charsets management
  - Lighter layout without most of the deprecated HTML elements and attributes
- Successfully compiled and installed on:
  - [x86] Linux 2.4 (Redhat 8.0)
  - [x86] Linux 2.4 (Redhat 7.3)
  - [x86] Linux 2.4 (Debian 2.2)
  - [x86] FreeBSD (4.7-STABLE)
  - [Alpha] Linux 2.4 (Debian 3.0)
  - [PPC - G4] MacOS X 10.1 SERVER Edition (statically linked)
  - [Sparc - Ultra60] Linux 2.4 (Debian 3.0)


Release notes for htcheck-1.2.0 - 16 Sep 2002
- added the 'store_url_contents' for storing the content of an HTML document
- added the Proxy Authorization support ('http_proxy_authorization')
- Keep trace of the bad encoded URLs through the 'url_reserved_chars' attribute
- Cookies are now handled as both the RFC2109 and Netscape say
- internal URLs are distinguished by external ones and the info is now stored
- HTML's 'id' attribute is now used for anchors, besides the 'name' attribute
- added the 'db_name_prepend' attribute for setting the string to
  be prepended to every database created by htcheck (also manageable
  through the 'with-db-name-prepend' configure option)
- added the 'remove_default_doc' attribute for removing the default document
  for a directory index
- added the '-k' feature for dropping just the tables, not the whole db
```

```
- Database changes:
  - New fields stored:
    - URL's content (Url)
    - HTML statement's row (HtmlStatement)
    - Server's IP address (Server)
    - Cookie version (Cookies)
- PHP Interface:
  - safer against XSS (cross-site scripting) attacks
  - Show the source of an HTML file
  - Filter for anchors now added to the links form
  - Added the support for 'tidy' (tidy.sourceforge.net) which allows to
    show the warning, errors and suggestions provided by this validator
- fixed some other minor bugs and made the code more robust


Release notes for htcheck-1.1 - 18 Feb 2002
- HTTP code now handles the language negotiation, through the
   'accept-language' attribute of the configuration file
- More robust support of cookies with the management of the domain attribute
- Cookies are now stored in the database (Cookies table)
- builds under GCC3
- fixed a bug regarding the BASE tag handling
- fixed some other minor bugs
- PHP Interface:
   - German language file added (thanks to Michael Stenitzer <stenitzer@eva.ac.at>)
   - some Web structure mining indexes have been added
   - display of the content language of a URL as given by the server
   - cookies simple report in the database home page
   - some cosmetic changes
   - code now has only the 'php' extension and works without the ASP tags setting


Release notes for htcheck-1.1.0b9-klunk - 25 Jun 2001
- Database structure now improved and compressed; less storage
  space and more speed in queries.
- Indexes of the Link table are created at the end of the crawl,
  improving performances, and controled by the 'url_index_length' parameter
- 'url_index_length' configuration attribute has been added: this
   attribute allows the user to control the length of the index
   for the Url field in the Schedule and Url tables. This attribute
   may affect the performance of the crawls, as long as the length
   of an index can either slow down or speed up the spidering process.
- Cookies summary (with -s option)
- POSIX standard: --version and --help compatible (with getopt_long)
- libtool 1.4 support
- fixed many bugs regarding the parser of the spider, which is now more robust
- cleaned code inside the 'core' source files
- PHP Interface:
   - Automatic and manual choosing of ht://Check databases
   - Javascript URLs query support
   - Description of a connection trouble when a URL is not retrieved
   - Fixed minor bugs and done cosmetic changes


Release notes for htcheck-1.1.0b8-muttley - 27 Apr 2001
```

```
- Finally runs on Solaris
- MySQL 3.23.xx users: now datetime fields are stored properly
- Link to e-mail are now stored and can be seen
- Link with a 'file:/' call are now considered as errors
- User Agent now shows the version and the platform
- Fixed a bug regarding the HTML parser with (very) malformed tags
- Fixed many minor bugs
- PHP Interface:
   - Enhancements: retrieve e-mail links
   - Fixed some bugs


Release notes for htcheck-1.1.0b7-anaconda - 28 Mar 2001
- Fixed library versioning
- Man page now provided (thanks to Marco Nenciarini <mnencia@prato.linux.it>
- Static linking now works fine
- New library architecture in order to provide no conflict with ht://Dig; they
  are all 'package' libs instead of global libs.
- 'optimize_db' has now been set to false by default
- PHP Interface:
   - PHP3 compatibility issued
   - removed .inc extension as PHP source


Release notes for htcheck-1.1.0b6-zizou - 12 Mar 2001
- HTTP Cookies support now enabled
- New type of link result: 'Not authorized'
- Fixed configuration error for load_mysql_defaults function and
  raised by Free BSD users.
- disable_cookies attribute added in the configuration
- Update of the HtDateTime class according to ht://Dig's one
- PHP interface:
  - better output
  - added images for link results
  - bug in qryurls.php and listlinks.php has been fixed
  - css file added for content visualization
  - dynamic language detection (english or italian for now)
- small bugs fixes


Release notes for htcheck-1.1.0b5-flukekelso - 24 Jan 2001
- Fixed a bug in the database initialization
- Default MySQL authentication (through /etc/my.cnf or ~/.my.cnf file)
- 'OBJECT' HTML tag now correctly parsed
- Basic HTTP Authentication enabled
- PHP interface improvements:
  - English and italian languages available
  - Get info regarding URLs by choosing through a form lots of parameters
    (i.e. URL, status code values, content-type, size and title if present)
   - Other small enhancements
- Documentation started
- Fixed other minor bugs


Release notes for htcheck-1.1.0b4-utero - 07 Sep 2000
```

```
- Now ht://Check uses MySQL's option file in order to get connection
  information such host, user, password, port and socket.
- HTTP Proxy support (to be tested more deeply)
- PHP interface's improvements:
  - It's now possible to look for broken links and anchors not found by
    using the form in listlinks.php. Filter can now be made with the
    LinkResult as well as the LinkType (and the referencing and referenced
    URLs like before).
- Fixed a bug regarding SGML entities with anchors and the "#top" anchor
  is now considered as valid.
- Sources have now been cleaned from most of the compilation warnings.


Release notes for htcheck-1.1.0b3-utero - 22 Aug 2000
- Better summary of the broken links (more complete and reliable).
- HTML anchors check is now performed and a field (LinkResult) has
  been added. It contains info about the link, if it's ok, broken,
  redirected and if a anchor is present and not found it warns about it.
- Summary of anchors not found, enabled or disabled through the
  configuration attribute 'summary_anchor_not_found'.
- The table 'htCheck' has been added to the database: its purpose
  is to store the general info of the crawl (user, start time, end
  time, etc ...).
- Added 'optimize_db' configuration parameter for optimizing the tables
  of the database. Default is true.
- Added 'sql_big_table_option' configuration parameter for performing
  huge queries. Default is true.
- Fixed the bug regarding HTTP persistent connections with a preemptive
  HEAD call before the GET.
- HTTP redirections are now treated as special links and stored into
  the link table with a 'Redirection' LinkResult flag.
- Referer management now is done right.
- Hop count management and storing added.
- Added 'max_hop_count' configuration parameter for limiting the crawl
  to a certain distance from the starting URL.
- PHP Interface:
  - The configure and make system has been modified in order to manage the
    php scripts. A new configuration option has been issued (--with-php-dir=DIR)
    and the make install procedure now look after the scripts too.
  - Page for querying the links retrieved, with a form which we
    can set filters through, regarding both the source and the
    destination URLs (with like and not like SQL statements);
  - Page for dropping a database.
  - Italian language added (include/italian.inc - See the INSTALL file)


Release notes for htcheck-1.1.0b2-utero - 08 Aug 2000
- A simple PHP interface has been added. You need PHP (either as a
  standalone CGI interpreter or - if you have Apache - as an Apache
  module) compiled with the mysql add-on module. For its installation
  look at the INSTALL file.
- The 'Link' table contains another field, the 'Anchor': its purpose
  is to store the 'token' after the '#' char in a link (for example in
  <A href="URL#anchorname">, it contains 'anchorname).
```

```
Release notes for htcheck-1.1.0b1-utero - 12 May 2000
A more stable version, but tested only on a RedHat 6.x system (see README file).
This new features have been added:
- Now it's possible to determine if a link is normal (like A href ones), that is
  to say the user has to click in order to get it, or is direct (like IMG src)
  that is to say it's automatically loaded (potentially) by the user's browser.
- Added a field to the Url table which contains the size to be added at load
  time in order to obtain the total weight of the document: it contains the sum


Release notes for htcheck-1.1.0b-utero - 5 May 2000
This is the very first release. It can be used for checking broken links.
Here are the main features:
- Access to a MySQL database (in this form: user@localhost, where user
  is the PID owner).
- HTTP 1.1 connections working with persistent connections choose
- At the end, show of broken links, servers seen and content-types encountered.
- Creation of these tables in the database: Url, Server, Link, Schedule,
  HtmlStatement, HtmlAttribute.
```

9 Copying (GNU General Public License)


```
                            Preamble

  The licenses for most software are designed to take away your
freedom to share and change it.  By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users.  This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it.  (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.)  You can apply it to
your programs, too.

  When we speak of free software, we are referring to freedom, not
price.  Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

  To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.
```

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

```
You may charge a fee for the physical act of transferring a copy, and
you may at your option offer warranty protection in exchange for a fee.

  2. You may modify your copy or copies of the Program or any portion
of it, thus forming a work based on the Program, and copy and
distribute such modifications or work under the terms of Section 1
above, provided that you also meet all of these conditions:

    a) You must cause the modified files to carry prominent notices
    stating that you changed the files and the date of any change.

    b) You must cause any work that you distribute or publish, that in
    whole or in part contains or is derived from the Program or any
    part thereof, to be licensed as a whole at no charge to all third
    parties under the terms of this License.

    c) If the modified program normally reads commands interactively
    when run, you must cause it, when started running for such
    interactive use in the most ordinary way, to print or display an
    announcement including an appropriate copyright notice and a
    notice that there is no warranty (or else, saying that you provide
    a warranty) and that users may redistribute the program under
    these conditions, and telling the user how to view a copy of this
    License.  (Exception: if the Program itself is interactive but
    does not normally print such an announcement, your work based on
    the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole.  If
identifiable sections of that work are not derived from the Program,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works.  But when you
distribute the same sections as part of a whole which is a work based
on the Program, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Program.

In addition, mere aggregation of another work not based on the Program
with the Program (or with a work based on the Program) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

  3. You may copy and distribute the Program (or a work based on it,
under Section 2) in object code or executable form under the terms of
Sections 1 and 2 above provided that you also do one of the following:

    a) Accompany it with the complete corresponding machine-readable
    source code, which must be distributed under the terms of Sections
    1 and 2 above on a medium customarily used for software interchange; or,
```

```
    b) Accompany it with a written offer, valid for at least three
    years, to give any third party, for a charge no more than your
    cost of physically performing source distribution, a complete
    machine-readable copy of the corresponding source code, to be
    distributed under the terms of Sections 1 and 2 above on a medium
    customarily used for software interchange; or,

    c) Accompany it with the information you received as to the offer
    to distribute corresponding source code.  (This alternative is
    allowed only for noncommercial distribution and only if you
    received the program in object code or executable form with such
    an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for
making modifications to it.  For an executable work, complete source
code means all the source code for all modules it contains, plus any
associated interface definition files, plus the scripts used to
control compilation and installation of the executable.  However, as a
special exception, the source code distributed need not include
anything that is normally distributed (in either source or binary
form) with the major components (compiler, kernel, and so on) of the
operating system on which the executable runs, unless that component
itself accompanies the executable.

If distribution of executable or object code is made by offering
access to copy from a designated place, then offering equivalent
access to copy the source code from the same place counts as
distribution of the source code, even though third parties are not
compelled to copy the source along with the object code.

  4. You may not copy, modify, sublicense, or distribute the Program
except as expressly provided under this License.  Any attempt
otherwise to copy, modify, sublicense or distribute the Program is
void, and will automatically terminate your rights under this License.
However, parties who have received copies, or rights, from you under
this License will not have their licenses terminated so long as such
parties remain in full compliance.

  5. You are not required to accept this License, since you have not
signed it.  However, nothing else grants you permission to modify or
distribute the Program or its derivative works.  These actions are
prohibited by law if you do not accept this License.  Therefore, by
modifying or distributing the Program (or any work based on the
Program), you indicate your acceptance of this License to do so, and
all its terms and conditions for copying, distributing or modifying
the Program or works based on it.

  6. Each time you redistribute the Program (or any work based on the
Program), the recipient automatically receives a license from the
original licensor to copy, distribute or modify the Program subject to
these terms and conditions.  You may not impose any further
restrictions on the recipients' exercise of the rights granted herein.
You are not responsible for enforcing compliance by third parties to
this License.
```

```
  7. If, as a consequence of a court judgment or allegation of patent
infringement or for any other reason (not limited to patent issues),
conditions are imposed on you (whether by court order, agreement or
otherwise) that contradict the conditions of this License, they do not
excuse you from the conditions of this License.  If you cannot
distribute so as to satisfy simultaneously your obligations under this
License and any other pertinent obligations, then as a consequence you
may not distribute the Program at all.  For example, if a patent
license would not permit royalty-free redistribution of the Program by
all those who receive copies directly or indirectly through you, then
the only way you could satisfy both it and this License would be to
refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under
any particular circumstance, the balance of the section is intended to
apply and the section as a whole is intended to apply in other
circumstances.

It is not the purpose of this section to induce you to infringe any
patents or other property right claims or to contest validity of any
such claims; this section has the sole purpose of protecting the
integrity of the free software distribution system, which is
implemented by public license practices.  Many people have made
generous contributions to the wide range of software distributed
through that system in reliance on consistent application of that
system; it is up to the author/donor to decide if he or she is willing
to distribute software through any other system and a licensee cannot
impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

  8. If the distribution and/or use of the Program is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Program under this License
may add an explicit geographical distribution limitation excluding
those countries, so that distribution is permitted only in or among
countries not thus excluded.  In such case, this License incorporates
the limitation as if written in the body of this License.

  9. The Free Software Foundation may publish revised and/or new versions
of the General Public License from time to time.  Such new versions will
be similar in spirit to the present version, but may differ in detail to
address new problems or concerns.

Each version is given a distinguishing version number.  If the Program
specifies a version number of this License which applies to it and "any
later version", you have the option of following the terms and conditions
either of that version or of any later version published by the Free
Software Foundation.  If the Program does not specify a version number of
this License, you may choose any version ever published by the Free Software
Foundation.

  10. If you wish to incorporate parts of the Program into other free
programs whose distribution conditions are different, write to the author
```

to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License

```
    along with this program; if not, write to the Free Software
    Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA  02111-1307  USA


Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

    Gnomovision version 69, Copyright (C) 19yy name of author
    Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type ‘show w’.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type ‘show c’ for details.

The hypothetical commands ‘show w’ and ‘show c’ should show the appropriate
parts of the General Public License.  Of course, the commands you use may
be called something other than ‘show w’ and ‘show c’; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary.  Here is a sample; alter the names:

  Yoyodyne, Inc., hereby disclaims all copyright interest in the program
  ‘Gnomovision’ (which makes passes at compilers) written by James Hacker.

  <signature of Ty Coon>, 1 April 1989
  Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs.  If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library.  If this is what you want to do, use the GNU Library General
Public License instead of this License.
```

10 Copyright

Copyright © 1999-2006 Comune di Prato - Prato - Italy
Some portions Copyright © 1995-2003 The ht://Dig Group

11 Thanks to ...

Of course I'd never have finished this program alone, all by myself: so I want to thank everybody who helped me, by giving me ideas, answers to my questions, and ... being sympathetic to me too!
Almost in a strict alphabetical order:

* Claudia Giorgetti, my director for really wanting this program!
* Francesca Becucci
* Valentino Bianco
* Fabrizio Butini
* Izak Burger

* Martina Ceccolini
* Loic Dachary
* Valentina Del Sapio for the PHP interface of the accessibility checks
* Gilles Detillieux
* Geoff Hutchison
* Robert LaFerla
* Sara Lenzi
* David Lippi
* Massimo Mango
* Nenciarini Marco <mnencia@prato.linux.it>
* Edward Moon
* Chad Picha
* Charlie Reitsma
* Michael Stenitzer

Forgive me if I forgot one of you. ;-)