

The Web Robots FAQ...

These frequently asked questions about Web robots.

Send suggestions and comments to [Martijn Koster](#). This information is in the public domain.

Table of Contents

1. About WWW robots
 - [What is a WWW robot?](#)
 - [What is an agent?](#)
 - [What is a search engine?](#)
 - [What kinds of robots are there?](#)
 - [So what are Robots, Spiders, Web Crawlers, Worms, Ants?](#)
 - [Aren't robots bad for the web?](#)
 - [Are there any robot books?](#)
 - [Where do I find out more about robots?](#)
 2. Indexing robots
 - [How does a robot decide where to visit?](#)
 - [How does an indexing robot decide what to index?](#)
 - [How do I register my page with a robot?](#)
 3. For Server Administrators
 - [How do I know if I've been visited by a robot?](#)
 - [I've been visited by a robot! Now what?](#)
 - [A robot is traversing my whole site too fast!](#)
 - [How do I keep a robot off my server?](#)
 4. Robots exclusion standard
 - [Why do I find entries for /robots.txt in my log files?](#)
 - [How do I prevent robots scanning my site?](#)
 - [Where do I find out how /robots.txt files work?](#)
 - [Will the /robots.txt standard be extended?](#)
 - [What if I cannot make a /robots.txt?](#)
 - [Surely listing sensitive files is asking for trouble?](#)
 5. Availability
 - [Where can I use a robot?](#)
 - [Where can I get a robot?](#)
 - [Where can I get the source code for a robot?](#)
 - [I'm writing a robot, what do I need to be careful of?](#)
 - [I've written a robot, how do I list it?](#)
-

About Web Robots

What is a WWW robot?

A robot is a program that automatically traverses the Web's hypertext structure by retrieving a document, and recursively retrieving all documents that are referenced.

Note that "recursive" here doesn't limit the definition to any specific traversal algorithm; even if a robot applies some heuristic to the selection and order of documents to visit and spaces out requests over a long space of time, it is still a robot.

Normal Web browsers are not robots, because they are operated by a human, and don't automatically retrieve referenced documents (other than inline images).

Web robots are sometimes referred to as Web Wanderers, Web Crawlers, or Spiders. These names are a bit misleading as they give the impression the software itself moves between sites like a virus; this is not the case, a robot simply visits sites by requesting documents from them.

What is an agent?

The word "agent" is used for lots of meanings in computing these days. Specifically:

Autonomous agents

are programs that do travel between sites, deciding themselves when to move and what to do. These can only travel between special servers and are currently not widespread in the Internet.

Intelligent agents

are programs that help users with things, such as choosing a product, or guiding a user through form filling, or even helping users find things. These have generally little to do with networking.

User-agent

is a technical name for programs that perform networking tasks for a user, such as Web User-agents like Netscape Navigator and Microsoft Internet Explorer, and Email User-agent like Qualcomm Eudora etc.

What is a search engine?

A search engine is a program that searches through some dataset. In the context of the Web, the word "search engine" is most often used for search forms that search through databases of HTML documents gathered by a robot.

What other kinds of robots are there?

Robots can be used for a number of purposes:

- [Indexing](#)
- HTML validation
- Link validation
- "What's New" monitoring
- Mirroring

See the [list of active robots](#) to see what robot does what. Don't ask me -- all I know is what's on the list...

So what are Robots, Spiders, Web Crawlers, Worms, Ants

They're all names for the same sort of thing, with slightly different connotations:

Robots

the generic name, see [above](#).

Spiders

same as robots, but sounds cooler in the press.

Worms

same as robots, although technically a worm is a replicating program, unlike a robot.

Web crawlers

same as robots, but note [WebCrawler](#) is a specific robot

WebAnts

distributed cooperating robots.

Aren't robots bad for the web?

There are a few reasons people believe robots are bad for the Web:

- Certain robot implementations can (and have in the past) overloaded networks and servers. This happens especially with people who are just starting to write a robot; these days there is sufficient information on robots to prevent some of these mistakes.
- Robots are operated by humans, who make mistakes in configuration, or simply don't

consider the implications of their actions. This means people need to be careful, and robot authors need to make it difficult for people to make mistakes with bad effects

- Web-wide indexing robots build a central database of documents, which doesn't scale too well to millions of documents on millions of sites.

But at the same time the majority of robots are well designed, professionally operated, cause no problems, and provide a valuable service in the absence of widely deployed better solutions.

So no, robots aren't inherently bad, nor inherently brilliant, and need careful attention.

Are there any robot books?

Yes:

[Internet Agents: Spiders, Wanderers, Brokers, and Bots](#) by Fah-Chun Cheong.

This books covers Web robots, commerce transaction agents, Mud agents, and a few others. It includes source code for a simple Web robot based on top of libwww-perl4.

Its coverage of HTTP, HTML, and Web libraries is a bit too thin to be a "how to write a web robot" book, but it provides useful background reading and a good overview of the state-of-the-art, especially if you haven't got the time to find all the info yourself on the Web.

Published by [New Riders](#), ISBN 1-56205-463-5.

[Bots and Other Internet Beasties](#) by Joseph Williams

I haven't seen this myself, but someone said: The William's book 'Bots and other Internet Beasties' was quite disappointing. It claims to be a 'how to' book on writing robots, but my impression is that it is nothing more than a collection of chapters, written by various people involved in this area and subsequently bound together.

Published by Sam's, ISBN: 1-57521-016-9

[Web Client Programming with Perl](#) by Clinton Wong

This O'Reilly book is planned for Fall 1996, check the [O'Reilly Web Site](#) for the current status. It promises to be a practical book, but I haven't seen it yet.

A few others can be found on the [The Software Agents Mailing List FAQ](#)

Where do I find out more about robots?

There is a Web robots home page on: <http://www.robotstxt.org/wc/robots.html>

Of course the latest version of this FAQ is there.

You'll also find details and an archive of the robots mailing list, which is intended for technical discussions about robots.

Indexing robots

How does a robot decide where to visit?

This depends on the robot, each one uses different strategies. In general they start from a historical list of URLs, especially of documents with many links elsewhere, such as server lists, "What's New" pages, and the most popular sites on the Web.

Most indexing services also allow you to submit URLs manually, which will then be queued and visited by the robot.

Sometimes other sources for URLs are used, such as scanners through USENET postings, published mailing list archives etc.

Given those starting points a robot can select URLs to visit and index, and to parse and use as a source for new URLs.

How does an indexing robot decide what to index?

If an indexing robot knows about a document, it may decide to parse it, and insert it into its database. How this is done depends on the robot: Some robots index the HTML Titles, or the first few paragraphs, or parse the entire HTML and index all words, with weightings depending on HTML constructs, etc. Some parse the META tag, or other special hidden tags.

We hope that as the Web evolves more facilities becomes available to efficiently associate meta data such as indexing information with a document. This is being worked on...

How do I register my page with a robot?

You guessed it, it depends on the service :-). Most services have a link to a URL submission form on their search page.

Fortunately you don't have to submit your URL to every service by hand: [Submit-it](http://www.submit-it.com/) <URL: http://www.submit-it.com/> will do it for you.

For Server Administrators

How do I know if I've been visited by a robot?

You can check your server logs for sites that retrieve many documents, especially in a short time.

If your server supports User-agent logging you can check for retrievals with unusual User-agent header values.

Finally, if you notice a site repeatedly checking for the file '/robots.txt' chances are that is a robot too.

I've been visited by a robot! Now what?

Well, nothing :-). The whole idea is they are automatic; you don't need to do anything.

If you think you have discovered a new robot (ie one that is not listed on [the list of active robots](#), and it does more than sporadic visits, drop me a line so I can make a note of it for future reference. But please don't tell me about every robot that happens to drop by!

A robot is traversing my whole site too fast!

This is called "rapid-fire", and people usually notice it if they're monitoring or analysing an access log file.

First of all check if it is a problem by checking the load of your server, and monitoring your servers' error log, and concurrent connections if you can. If you have a medium or high performance server, it is quite likely to be able to cope a high load of even several requests per second, especially if the visits are quick.

However you may have problems if you have a low performance site, such as your own desktop PC or Mac you're working on, or you run low performance server software, or if you have many long retrievals (such as CGI scripts or large documents). These problems manifest themselves in refused connections, a high load, performance slowdowns, or in extreme cases a system crash.

If this happens, there are a few things you should do. Most importantly, start logging information: when did you notice, what happened, what do your logs say, what are you doing in response etc; this helps investigating the problem later. Secondly, try and find out where the robot came from, what IP addresses or DNS domains, and see if they are mentioned in the [list of active robots](#). If you can identify a site this way, you can email the person responsible, and ask them what's up. If this doesn't help, try their own site for telephone numbers, or mail postmaster at their domain.

If the robot is not on the list, mail me with all the information you have collected, including actions on your part. If I can't help, at least I can make a note of it for others.

How do I keep a robot off my server?

Read the next section...

Robots exclusion standard

Why do I find entries for /robots.txt in my log files?

They are probably from robots trying to see if you have specified any rules for them using the [Standard for Robot Exclusion](#), see also [below](#).

If you don't care about robots and want to prevent the messages in your error logs, simply create an empty file called robots.txt in the root level of your server.

Don't put any HTML or English language "Who the hell are you?" text in it -- it will probably never get read by anyone :-)

How do I prevent robots scanning my site?

The quick way to prevent robots visiting your site is put these two lines into the /robots.txt file

on your server:

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

but its easy to [be more selective](#) than that.

Where do I find out how /robots.txt files work?

You can read [the whole standard specification](#) but the basic concept is simple: by writing a structured text file you can indicate to robots that certain parts of your server are off-limits to some or all robots. It is best explained with an example:

```
# /robots.txt file for http://webcrawler.com/  
# mail webmaster@webcrawler.com for constructive criticism  
  
User-agent: webcrawler  
Disallow:  
  
User-agent: lycra  
Disallow: /  
  
User-agent: *  
Disallow: /tmp  
Disallow: /logs
```

The first two lines, starting with '#', specify a comment

The first paragraph specifies that the robot called 'webcrawler' has nothing disallowed: it may go anywhere.

The second paragraph indicates that the robot called 'lycra' has all relative URLs starting with '/' disallowed. Because all relative URL's on a server start with '/', this means the entire site is closed off.

The third paragraph indicates that all other robots should not visit URLs starting with /tmp or /log. Note the '*' is a special token, meaning "any other User-agent"; you cannot use wildcard patterns or regular expressions in either User-agent or Disallow lines.

Two common errors:

- Wildcards are not supported: instead of 'Disallow: /tmp/*' just say 'Disallow: /tmp'.
- You shouldn't put more than one path on a Disallow line (this may change in a future version of the spec)

Will the /robots.txt standard be extended?

Probably... there are some ideas floating around. They haven't made it into a coherent proposal because of time constraints, and because there is little pressure. Mail suggestions to the robots mailing list, and check the robots home page for work in progress.

What if I can't make a /robots.txt file?

Sometimes you cannot make a /robots.txt file, because you don't administer the entire server. All is not lost: there is a new [standard](#) for using HTML META tags to keep robots out of your documents.

The basic idea is that if you include a tag like:

```
<META NAME="ROBOTS" CONTENT="NOINDEX" >
```

in your HTML document, that document won't be indexed.

If you do:

```
<META NAME="ROBOTS" CONTENT="NOFOLLOW" >
```

the links in that document will not be parsed by the robot.

Surely listing sensitive files is asking for trouble?

Some people are concerned that listing pages or directories in the /robots.txt file may invite unintended access. There are two answers to this.

The first answer is a workaround: You could put all the files you don't want robots to visit in a separate sub directory, make that directory un-listable on the web (by configuring your server), then place your files in there, and list only the directory name in the /robots.txt. Now an ill-willed robot can't traverse that directory unless you or someone else puts a direct link on the web to one of your files, and then it's not /robots.txt fault.

For example, rather than:

```
User-Agent: *  
Disallow: /foo.html  
Disallow: /bar.html
```

do:

```
User-Agent: *  
Disallow: /norobots/
```

and make a "norobots" directory, put foo.html and bar.html into it, and configure your server to not generate a directory listing for that directory. Now all an attacker would learn is that you have a "norobots" directory, but he won't be able to list the files in there; he'd need to guess their names.

However, in practice this is a bad idea -- it's too fragile. Someone may publish a link to your files on their site. Or it may turn up in a publicly accessible log file, say of your user's proxy server, or maybe it will show up in someone's web server log as a Referer. Or someone may misconfigure your server at some future date, "fixing" it to show a directory listing. Which leads me to the real answer:

The real answer is that /robots.txt is not intended for access control, so don't try to use it as such. Think of it as a "No Entry" sign, not a locked door. If you have files on your web site that you don't want unauthorized people to access, then configure your server to do authentication, and configure appropriate authorization. Basic Authentication has been around since the early days of the web (and in e.g. Apache on UNIX is trivial to configure), and if you're really serious, SSL is commonplace in web servers.

Availability

Where can I use a robot?

If you mean a search service, check out the various directory pages on the Web, such as Netscape's [Exploring the Net](#) or try one of the Meta search services such as [MetaSearch](#)

Where can I get a robot?

Well, you can have a look at the list of robots; I'm starting to indicate their public availability slowly.

In the meantime, two indexing robots that you should be able to get hold of are Harvest (free), and Verity's.

Where can I get the source code for a robot?

See [above](#) -- some may be willing to give out source code.

Alternatively check out the libwww-perl5 package, that has a simple example.

I'm writing a robot, what do I need to be careful of?

Lots. First read through all the stuff on [the robot page](#) then read the proceedings of past WWW Conferences, and the complete HTTP and HTML spec. Yes; it's a lot of work :-)

I've written a robot, how do I list it?

Simply fill in a form you can find on [The Web Robots Database](#) and email it to me.

[The Web Robots Pages](#)