

Website

Mirror

Guide

by Web
Hosting
Geeks

2012

Website Mirror

Guide

Description

The purpose of this guide is to explain how to create mirror systems for various software. It also presents the main difficulties encountered in the process of setting up a mirror system and ways of overcoming them.

The given guide is intended for system administrators and SEO specialists.

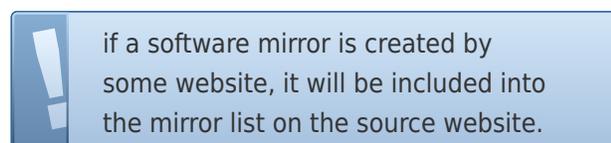
Contents

1.	Introduction	4
2.	Creating an Address	4
	2.1. Creating a Mirror for Your Own Needs	4
	2.2. Creating a Mirror for a Partner	4
3.	Estimated Disk Space	5
	3.1. General Considerations	5
	3.2. Preliminary Estimation of Mirror Size	6
4.	Creating a Mirror	9
	4.1. Creating a Mirror Based on rsync	9
	4.2. Creating an FTP Based Mirror	10
	4.3. Monitoring Disk Space	10
5.	Adding a Mirror to a Mirror List	11
	5.1. Option 1	11
	5.2. Option 2	11
	5.3. Option 3	11
	5.4. Letter Contents	12
	5.5. Signature and Address	12
6.	Mirror Distribution	13
7.	Organizing Work Procedure	13
8.	Conclusions	14
9.	About the Author	16
10.	Colophon	16
11.	About Web Hosting Geeks	16

1. Introduction

The purpose of creating a mirror system is to obtain links from websites with a high PR.

As a rule, well-known websites elaborated by software developers have a significantly high PR. At the same time, in order to reduce the load on the server from which end users download the software, they create mirrors for their software (for small content - a complete mirror of the site, for large amounts of content - mirrors of file repositories). Without going into the historical motivation of this phenomenon we should point out only one thing -



Thereby, its PR will automatically increase.

2. Creating an Address

2.1. Creating a Mirror for Your Own Needs

If you set a mirror for your own needs there's no problem - you create an ordinary website, where each mirror is placed in its own directory.

2.2 Creating a Mirror for a Partner

The easiest way to set up a mirror for a partner is to create a CNAME record indicating your own mirror's address as your partner's mirror address.

For example: your partner's website is **domain.com** and your own is **mirrors.wikipedia.org**.

Create a mirror - **mirrors.domain.com** - based on your website **mirrors.wikipedia.org**.

Setting up a software mirror system includes the following steps:

- › Creating an address
- › Allocating required disk space
- › Creating mirrors
- › Adding mirrors to mirror list.

This document contains materials dealing with setting up mirror systems based on the website mirrors.wikipedia.org.

The purpose of elaborating this document is to present the work that has been done in order to analyze the results, identify the key problems while creating mirror systems and develop suggestions how to solve them.

This document contains the main results of the work that has been carried out for four months, from September to December 2011.

To set up a mirror your partner must register a record on his DNS server: **mirrors.domain.com CNAME mirrors.mirrors.wikipedia.org**.

Next, in your web-server's settings, create a record for the website mirrors.partner.com specifying the root directory of mirrors.wikipedia.org as the root directory.

If necessary, we can similarly create mirrors as individual sub-domains; however, in this case, you should register a separate CNAME record for each sub-domain, for example: **apache.domain.com CNAME apache.mirrors.wikipedia.org**

It goes without saying that each record requires a special set of settings on your own web-server.

3. Estimated Disk Space

3.1. General Considerations

One of the key issues of creating a mirror system is sufficient amount of disk space for its installation. If you know how much disk space required, this issue can be easily solved. But how can you have any idea about the possible required capacity of the hard drive before you really start synchronizing your first project?

In an ideal case you do have the requirements and you know what to do. Unfortunately, our case was a bit different. Due to lack of experience (both ours and others') in this area, we started the mirror system with quite a reduced amount of disk space - 250 GB. Soon came the realization that that much was obviously not enough, so disk space was extended to 500 GB, by connecting a second hard drive of the same capacity. Though OS Linux with the use of LVM (Linux Volume Manager) allows to perform this operation safely, for some time there were cases of failure in work (up to a full stop of the server), which most likely happened due to the unstable performance of the file system.

Next, the real size of the allocated data is quite considerable. The total size of repositories of certain operating systems - Linux, various versions of BSD, which include different versions of OS (including the obsolete and unsupported ones, binary and source codes, software packages, etc.) - can easily reach up hundreds or more GB. That's why we had to change our understanding of the required disk capacity once again.

Date	Disk capacity	Number of mirrors
July 15, 2011	250 GB	6
August 16, 2011	500 GB	9
September 09, 2011	2000 GB	12

Table 1. Evolution of hard disk capacity.

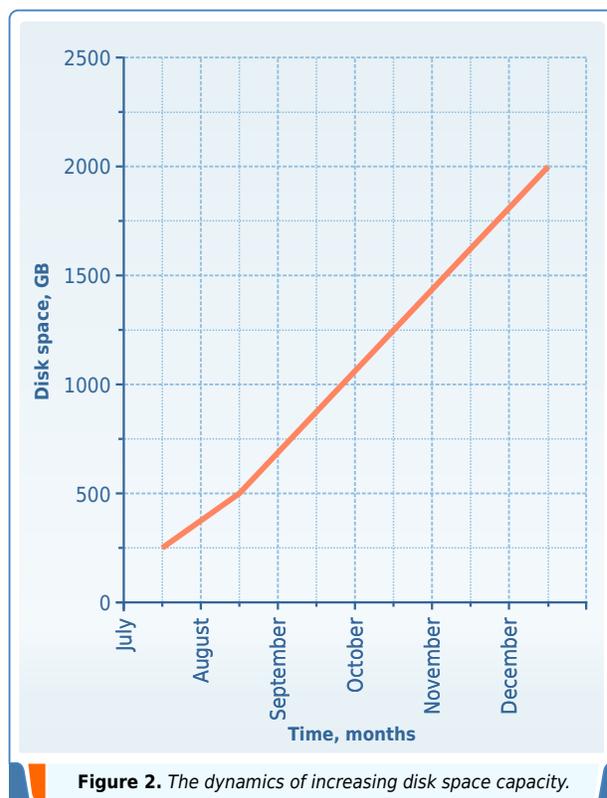


Figure 2. The dynamics of increasing disk space capacity.

In early December we decided to increase the amount of disk space up to not less than 1TB (the cost of dedicated server increased by \$25 per month). At the same time we clearly understood that the data had to be transferred on a single hard disk having that capacity, because adding another disk to it, of - for example - 500 GB - would severely reduce server reliability. Three disks running at the same time increases the probability of work failures. At the same time we had to take into account the possibility of further expanding disk space.

Note that using RAID wasn't planned, though in case of data loss it could be recovered within 1-3 days (for mirrors with large amounts of data) from the source websites.

The speed of recovery depends on the channel bandwidth and server load – the source and our own server. This reduces the total cost of equipment. However, you should pay great attention to the data security of your own services.

Being concerned with the question “to what point 1TB will be enough?” an agreement was reached concerning the installation a 2 TB hard drive, as the host had in stock such capacity HDD. Yet, in this case the cost of dedicated server would rise by \$35 per month. However, during the process of transferring data to a 2TB disk we experienced significant difficulties: this might have happened due our host not being qualified enough. The initial adding of a 2TB disk with transferred data ended up in failure. Our host motivated it by the fact that CentOS 5.6, installed on our server, can't provide loading data from a 2TB disk. So, we installed CentOS 6.1 and transferred the data. As a number of services were set up to work with CentOS 5.6, we had to perform some additional operations after the installation, such as:

- Restoring phpMyAdmin and its interaction with ISP manager;
- Restoring ISP manager functionality when editing files;

- › Restoring cron jobs;
- › Restoring backup settings;
- › Restoring the monitoring system.

All these constant connections and disconnections of the disks certainly destabilize work and ultimately reduce productivity, so, the entire process may take several weeks. Therefore, as a recommendation, mind the fact that you should count the initial capacity of disk space in the purpose of installing a mirrors system as much as 1TB. Thus, you'll avoid unnecessary trouble in the future, like the ones mentioned above.

3.2. Preliminary Estimation of Mirror Size

If you're not worried about disk space (or so it may seem to you), you may skip this step. However, in real life there's no such thing as unlimited resources, so we strongly recommend that you assess the mirror size before installing it.

First of all, search the website of the software owner. See if there's any indication of disk space amount necessary for a mirror. You can find this information in the instructions for mirror installation. If the instructions are missing, or these data are not there at all, you'll have to estimate the required capacity yourself.

The simplest way to do it is to gradually download the software you plan to install as a mirror via FTP from the source-website. In this case you have the possibility of gradually creating a mirror (by directories) while controlling the capacity.

In some cases an initial download using rsync by running it from the command line is possible. However, given the large size of mirrors, it is best to monitor the disk space in order to interrupt the download if necessary.

Be careful – the initial download of the mirror may take more than a day! The ideal solution for this is having a system of monitoring the server.

Actually, there are two other options for estimating the size of the mirror, which are less time consuming and less threatening to overload the disk.

The first option is to get information from the data owner, yet in some cases this is not always acceptable.

The second option is to use an utility that allows evaluating (count) the total size of mirror files without downloading them using any protocol (HTTP, FTP). Unfortunately, we couldn't find any software solutions for this problem, so we had to develop a php script for evaluating mirror sizes on a remote FTP-server. It use significantly reduced the time required to install the mirrors.

Here is the code of the script we used:

```

01. <?php
02. //=====
03. // Directory size
04. // #php path_to_script host_name directory
05. //
06. //=====
07. function ftp_dir_size($connect, $dir)
08. {
09. $dir_size=0;
10. $file_list = ftp_rawlist($connect, $dir);
11. //print_r ($file_list);
12. // get directory list
13. foreach($file_list as $file)
14. { $dim=explode(' ', $file);
15.   if (count($dim)>3)
16.     { //echo "-----\n";
17.
18. list($attr,$bloks,$group,$user,$size,$month,$day,$
19. year,$f_name)
20.         = preg_split("/[\s]+/", $file);
21.         $pr_dir=substr($attr, 0, 1);
22.         //echo $attr."\n";
23.         //echo $pr_dir."\n";
24.         //echo $f_name."\n";
25.         //echo $size."\n";
26.
27.         if (substr($file, 0, 1) != '.')
28.           { // directory
29.
30.             if($pr_dir == 'd')
31.               {$t_dir=$dir."/".$f_name."/";

```

```

32. $dir_size=$dir_size+ftp_dir_size($connect,
33.     $t_dir);
34.   }
35.   // file
36.   if($pr_dir == '.')
37.
38.     { echo "****".$f_name."-".$size."\n";
39.
40.     $dir_size=$dir_size+$size;
41.   }
42.
43. }
44.   }
45. }
46. return $dir_size;
47. }
48. //=====
49. set_time_limit(3600);
50. $host=$argv[1];
51. $dir=$argv[2];
52. echo $host."\n";
53. echo $dir."\n";
54. $user = "anonymous";
55. $password = "";
56. $connect=ftp_connect($host);
57. if ($connect)
58.   {$login = ftp_login($connect, $user, $password);
59.   if ($login)
60.     {if (ftp_chdir($connect, $dir))
61.       {$dir=ftp_pwd($connect);
62.       echo "new directory-".$dir."\n";
63.     }
64.     else {echo "Cannot change directory\n"; }
65.
66.     $size=ftp_dir_size($connect, $dir);
67.     // print directory size
68.     echo "\nDirectory size=".$size;
69.   }
70.   ftp_close($connect);
71. }
72. else {echo "Not connect with ".$host; }
73. ?>

```

In order to estimate the disk space occupied by mirrors on the server, there's a convenient utility - `ncdu`, which allows you to display a summary for each directory:

```

/..
175.6GB /dragonflybsd
141.7GB /centos
 58.0GB /cran
 27.3GB /gcc
 23.1GB /mozilla
 20.9GB /kde
 13.1GB /opera
 10.9GB /cpan
  4.1GB /imagemagick
  2.4GB /wp
 374.2MB /openssl
 271.7MB /xemacs
  30.4MB /email
  17.4MB /putty
  14.6MB /apache
    
```

Figure 3. Directory summary in `ncdu`.

It is recommended to put all the results in a table, which would allow to estimate the change (or stability) dynamics for each mirror.

	A	B	C	D
1				
2	December 01, 2011	December 02, 2011	December 03, 2011	December 04, 2011
3	207.7GB /dragonflybsd	191.0GB /dragonflybsd	194.2GB /dragonflybsd	185.1GB /dragonflybsd
4	124.6GB /centos	124.7GB /centos	124.7GB /centos	124.7GB /centos
5	22.8GB /mozilla	27.3GB /gcc	27.3GB /gcc	27.3GB /gcc
6	19.9GB /kde	22.9GB /mozilla	22.9GB /mozilla	22.9GB /mozilla
7	12.9GB /opera	19.9GB /kde	19.9GB /kde	19.9GB /kde
8	11.3GB /gcc	12.9GB /opera	12.9GB /opera	12.9GB /opera
9	4.0GB /imagemagick	4.0GB /imagemagick	4.0GB /imagemagick	4.0GB /imagemagick
10	486.0MB /wp	1.0GB /wp	1.0GB /wp	1.0GB /wp
11	374.7MB /openssl	374.7MB /openssl	374.7MB /openssl	374.7MB /openssl
12	275.4MB /xemacs	275.4MB /xemacs	275.4MB /xemacs	275.4MB /xemacs
13	30.6MB /email	30.6MB /email	30.6MB /email	30.6MB /email
14	17.5MB /putty	17.5MB /putty	17.5MB /putty	17.5MB /putty

Figure 4. Mirror capacities - an evaluation of change dynamics.

4. Creating a Mirror

4.1. Creating a Mirror Based on rsync

As a rule, the most common way to create a mirror is to synchronize the source-website from the rsync-server using an utility with the same name (rsync). If on software owner's website there are instructions for setting up rsync, then there are no problems whatsoever. For example, <http://cran.r-project.org/> provides detailed instructions on installing mirrors. As for setting up rsync, the following indications are given:

All you have to do is recursively mirror the complete tree to your webserver on a regular basis (at least twice a week, better every 1-2 days). Which software you use for mirroring depends on the operating system of your server, but we strongly recommend that you use [rsync](#). You may want to call it using the following arguments:

```
rsync -rtlv --delete cran.r-
project.org::CRAN/dir/on/local/disc
```

Do not forget the `--delete` flag to remove files from the mirror that are no longer present on the master.

Figure 5. *Rsync mirror setup directions.*

This set of directions contains at least two instances, that we're interested in:

- an address to download the data - `cran.r-project.org::CRAN` - the most important thing
- mirror update frequency - at least twice a week, but every 1-2 days is better.

According to these instructions we create a job in cron in order to run rsync; thus the process is complete. It is recommended to create logs of the rsync work, which, if necessary, would allow to clarify the reasons of synchronization failure.

Complete rsync documentation can be found on the website [Rsync](#). The documentation section ([click here](#)) shows all possible cases of using rsync.

An important moment should be pointed out. If there's insufficient disk space, it's preferable to use the `-delete` directive instead of `-delete-after`. In case of using the `-delete-after` directive, files are first downloaded, and the the missing ones are deleted on the source-website. In this case, the mirror will be quickly brought in full compliance with the original, which can be important for frequently changing mirrors, such as mozilla. In case of using the `-delete` directive, first, the missing files on the website are deleted, then the new ones are downloaded. This mode is recommended for large-sized mirrors (CentOS, DragonflyBSD).

Other adjustments depend on your preference, only if they don't contradict the above-mentioned directions.

In certain cases the software owner's website doesn't have any information about configuring mirrors ([click here](#)) and you might encounter a message such as:



If you wish to host a new mirror site, please contact gcc@gcc.gnu.org.

In this case, mirror installation is impossible without prior contact with the software owner (for more details see section 4).

4.2. Creating an FTP Based Mirror

A less frequent case is creating a FTP-based mirror. Actually, the setting procedure is the same as the one described above, except the fact that for synchronization you should use some FTP-client instead of rsync. Please note that the wget utility (as well as many others) isn't appropriate in our case, despite the fact that it is well-proven. This can be motivated by the fact that alongside with downloading files we need to remove the missing ones on the source-website as well. Perhaps, the best option is to use the lftp utility ([click here](#)) in the mirror mode.

You can find instructions on how to configure the lftp utility (including the mirror-mode) on its developer's website ([click here](#)).

4.3. Monitoring Disk Space

Once the mirrors are installed, it is recommended to install a system that would monitor available disk space on the server and send notifications (via e-mail or sms) in emergency cases. Thus, it will help avoiding server halt in the case of lack of disk space.

It's obvious that the issue of controlling disk space while having a large number of mirrors should be given considerable attention. Unlike software, any mirror can grow in size quite unexpectedly, for example, when a new version is released. In order to control disk space automatically, using a monitoring system is recommended. A good example is [Munin](#). It's quite a customizable and expandable system. Alongside with the possibility of controlling parameters visually, via web-interface, it also allows you to set up e-mail notifications when the controlled settings reach the threshold value.

In some cases, the size of the mirror can be reduced by synchronizing only stable versions of the software. Of course, it should be agreed upon

with the owner of the software. In this case mirror installation becomes more complicated; however, the benefit is obvious.

5. Adding a Mirror to a Mirror List

Adding a mirror to the mirror list on the software owner’s website, from my own point of view, is the most difficult process in the entire technology of mirror organization. This can be explained by the fact that all previously described options can be performed on your own. But in order to add a mirror to the list, you must obtain the permission of the website owner.

5.1. Option 1

In the most common case the instructions for installing mirrors may contain the following:

Install the mirror, write us at mirrors@example.com and we’ll add you to our mirror list.

Such was the case during the installation of the following mirrors:

- [ImageMagick](#);
- [KDE](#);
- [Dragonfly BSD](#).

However, even in this case, success should not be taken for granted. Perhaps the letter didn’t reach its recipient. Perhaps someone was going to do this, but couldn’t manage at that very moment, and then the task was forgotten. There are a thousand and one reasons for answer delays.

Here’s an example of correspondence in the purpose of creating a mirror for ImageMagick:

- On November 7, 2011, we sent out a request to create. We got a response on the same day: “Ok. When it’s ready, let us know and we will include it in our mirror web page”;

- On November 8, 2011, the mirror was set up and we sent a letter requesting its inclusion in the mirror list.
- No answer was received and therefore on November 9, 2011, we sent another letter with the same request.
- Only on November 17, 2011, we got a response: “Done. Look for it on the ImageMagick download page in about 24 hours. Thanks”.
- On November 17, 2011 the link was present in the [ImageMagick mirror list](#) – then it was verified and after confirming everything, the work was closed.

Anyway, if no response is received, I recommend sending letters requesting inclusion into the mirror list at least **three** times – that’s the maximum (speaking from personal experience) after which you may receive a reply to your letter.

5.2. Option 2

In some cases, the instructions for installing mirrors contain the following text:

Install the mirror, send us a letter and we’ll consider your inclusion in the mirror list.

Thus, there’s a possibility that your mirror won’t be included in the list. Nonetheless, you shouldn’t give up before trying. Just proceed according to steps described in Option 1.

5.3. Option 3

In some cases, in order to apply for inclusion to the mirror list, a subscription to the mailing list is required. For example, that was the case of [CentOS](#). It means that you should add your requests to the mailing list.

Here's an example of our correspondence with CentOS:

- The first letter was sent on September 30, 2011. No response was received.
- The second letter was sent on November 09, 2011. We received an answer which said that we should write to CentOS-mirror mailing list.
- We completed the registration in CentOS-mirror mailing list. A letter was written to the mailing list on November 17, 2011. No response was received.
- On November 19, 2011, another inquiry was written and on November 21, 2011, we received an answer that the mirror was added.

5.4. Letter Contents

So, you are going to send an email to the software owner with either a question concerning the possibility of creating a mirror and clarifying technical details, or for with a request for inclusion to the mirror list, when the mirror is already created.

In any case, you must have in the email the following minimum data set:

- Name of the organization
- Website address
- Name and email of the administrator of the website that hosts the mirror
- Geographical location of the server (this could be important) - country, city
- IP-address of the server
- Download protocol for the created mirror (HTTP, FTP) and its address. See the link samples below: [link 1](#) and [link 2](#).
- Maximum server connection speed.

Before sending the letter, read the instructions for mirror installation (if available) once again. In some cases, software owners ask to mention, for example, the maximum number of connections to

your server, or the number of synchronization processes. As a rule, it's enough to synchronize once a day, but sometimes you're asked to do it twice a day.

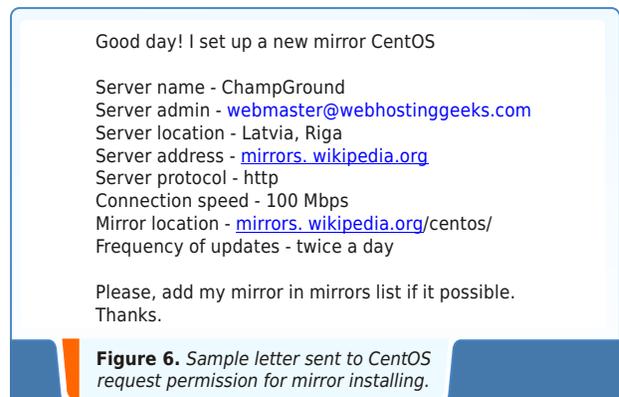
Be sure to have this information in your email.

5.5. Signature and Address

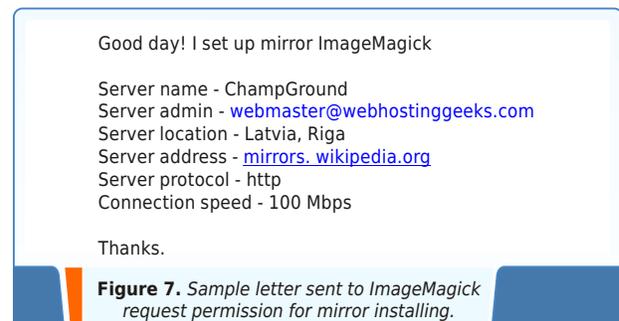
A letter signature must contain you name, position, e-mail address.

Position: it'd be perfect if you indicated that you are an administrator. In this case, the email address is important! It is better if the address domain matches the mirror domain. Or, as an exception to the generally accepted rules, the email account may be registered on Gmail. Having a Gmail email for the server administrator is regular practice.

Here's an example of a real letter (sent while installing a mirror on CentOS).



Here's another sample of a real letter - this one was sent while installing a mirror on ImageMagick.



6. Mirror Distribution

Given the fact that as a rule there's a web-server on almost any software server, we can organize the distribution of our mirrors using HTTP.

Nevertheless, we recommend to make appropriate settings for the FTP-server in order to distribute the mirrors via FTP.

In this case, at a low cost (installation and configuration of the FTP-server) we get at the least the following benefits:

- › We get two addresses in the mirror list on the source website.

- › The load on the web-server (which can be used for internal purposes) decreases due to partial transfer of outgoing traffic onto FTP-protocol.
- › Credibility of our website increases due to the fact that it is considered able to set up a fully functional mirror, which gives us more chances to be added to the mirror list.

Also, as additional means of reducing the load on the regular web-server, we may recommend installing an additional web-server for maintaining your own services, for example, *lighttpd*.

7. Organizing Work Procedure

Organizing a system of mirrors is quite a long process. Installing one mirror including the correspondence can last for weeks (by installation we mean the ultimate goal - adding the mirror to the mirror list on the source-website). Not to say that creating a system may turn out into a process that can last for months.

With this, the works on your own server must match with the process of correspondence. In this case, the process of creating mirrors is usually made for multiple source-websites. With this, it would be best to systematize the planning and monitoring of work performance ([click here](#)).

This document should contain at least the following information:

- › code name of the project;
- › website address;

- › address of the mirror list on the website;
- › address with the instruction on installing a mirror.

For each mirror, a graph indicating the state of work and planned operations should be provided.

It's necessary to log the correspondence on organizing the mirrors. It's up to you to decide whether you do it in the same document or create separate documents for each mirror. The following information should be provided - text of your letter and the date when it was sent, text of the reply and the date of receipt, e-mail address of the sender and of the recipient.

This will allow to carry out the process of creating mirrors in a more organized way.

8. Conclusions

In 4 months, the PR value of the domain **mirrors.wikipedia.org** was raised from 0 to PR=4.

Table 2 shows the data by websites, on which mirrors are installed and included to the mirror list.

Project name	Address	Q, GB	PR	Q/PR
Mozilla	www.mozilla.org	23	9	2.6
Opera	www.opera.com	13	9	1.4
KDE	www.kde.org	21	7	3.0
ImageMagick	www.imagemagick.org	4	7	0.6
Dragonfly BSD	www.dragonflybsd.org	177	6	29.5
CentOS	www.centos.org	142	7	20.0

Table 2. Data analysis by website that obtained mirrors.

Note: Q - disk space, occupied by the mirror on December 15, 2011.

The total size of occupied disk space shown in Table 2 is 380 GB.

The last column of the table shows the ratio Q/PR - a measure for mirror quality. The lower it is (relatively, of course), the more appropriate the mirror is for use.

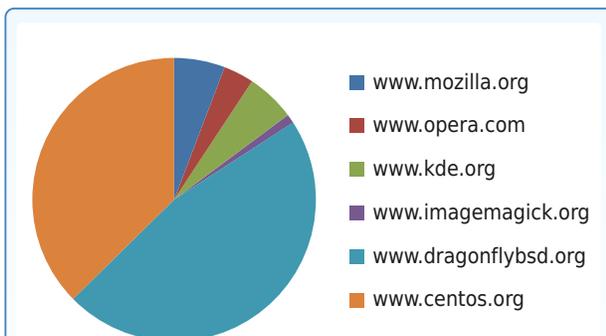


Figure 8. Disk space occupied by mirrors of popular website.

On the basis of the results from the table, we may conclude:

1 Placing operating systems as mirrors is quite expensive.

2 Creating a mirror on their basis is useful, if there's an agreement on limited deployment, for example for only stable product copies.

Let's take as an example Dragonfly BSD: its full size is approximately 400 GB. A particularly large amount of disk space was occupied by software packages for different versions. Because of this, the synchro-nization of the software was limited at the latest versions of Dragonfly BSD. As a result, the required disk space has been reduced at least by half, to a reasonable 200 GB.

Index of /dragonflybsd/packages/amd64

Name	Last modified	Size
Parent Directory		-
DragonFly - 2.12.0/	10-Nov-2011 02:11	-
DragonFly - 2.12.1/	10-Nov-2011 02:11	-
DragonFly - 2.12/	10-Nov-2011 02:11	-
DragonFly - 2.12.0/	15-Oct-2011 19:56	-
DragonFly - 2.13.1/	15-Oct-2011 19:56	-
DragonFly - 2.13/	15-Oct-2011 19:56	-

Figure 9. List of /dragonflybsd/packages/amd64 files.

However, achieving such an agreement in each case is likely impossible. Therefore, the problem of disk space will directly depend on the objectives set to increase the PR.

3 For the installation, you should use system-wide software, that works on different operating systems – web-servers, FTP-servers (ProFTPd), database servers (MySQL), etc.

Based on the analysis of Table 2, we adjusted the selection of mirrors we plan to install. Currently the mirrors presented in Table 3 are in the process of being installed (in different states of readiness, as the work has been stopped due to the change of the hard-disk).

Project name	Address	Q, GB	PR	Q/PR
apache	www.apache.org	28	9	3.1
cpan	www.cpan.org	10	7	1.4
cran	cran.r-project.org	74	7	10.6
gcc	gcc.gnu.org	27	7	3.9
openssl	www.openssl.org	0,374	8	0.05
putty	www.putty.org	0,017	6	0.003
xemacs	www.xemacs.org	0,071	6	0.012

Table 3. Mirrors being currently installed.

It's obvious that PR4 is not the final target.

These are the aspects that are costly:

- › providing the server with quite an extensive disk space capacity
- › paying the part-time server administrator

Also, you should remember that alongside with supporting a mirror system, a server is used for other purposes as well, which can generally reduce the cost of maintaining the mirror system. The total expenses per month rise to approximately \$500.

9. About the Author

Vladimir Tutubalin is a programmer in the Ephemeral Office.

Ecaterina Albu: proofreading and editing.

Kirill Belyaev: cover design.

Sergey Mikhailov: layout.

Colophon

The booklet is set in [DejaVu Sans Condensed](#), [CommonBullets](#).

Headlines are set in [League Gothic](#).

The cover is set in: [Alfa Slab Font](#).

About Web Hosting Geeks

Making the right choice when it comes to choosing a webhosting is never an easy task, neither for a rookie, nor for a pro. Our concise guides revolve around the all-encompassing topic of webhosting, providing solutions and points of view that you might not think of. We strive to boldly go where few dared to venture in matters of detail and perspective. Our geek mission is to highlight these solutions and serve them to you, for whatever site you build—a tiny personal blog, an important volunteer project or a sizeable business.