

« Previous    Next »



# µTorrent User Manual

v3.3.2 ( Last updated on: 17-August-2013, by: Rafi )

Welcome to the µTorrent User Manual. We hope the information provided here will assist you in using µTorrent.

You can always find the newest released version of µTorrent on the [downloads page](#).

If you have a question that isn't answered in this manual, please check [the forums](#) for related discussions, or submit a post of your own.

To view a **web-based version** of these help documents, please visit [this page](#).

# Table of Contents

- **Chapter 01: Introduction to  $\mu$ Torrent**
  - About  $\mu$ Torrent
  - Features List
- **Chapter 02: Basic Guides**
  - The Basics of BitTorrent
  - Setup Guide
  - Port Forwarding
  - Downloading With  $\mu$ Torrent
  - Downloading Using the RSS Downloader
- **Chapter 03: Advanced Guides**
  - More Port Forwarding
  - Migrating to Another Location
  - Switching from Another Client
  - Alternative Operating Systems
- **Appendix A: The  $\mu$ Torrent Interface**
  - Main Window
    - Main Menus
    - Toolbar
    - Category List
    - Torrent Jobs List
    - Detailed Info Pane
      - Files
      - Info
      - Ratings
      - Trackers
      - Peers
      - Pieces
      - Speed
      - Logger
      - Related
    - Status Bar
  - Preferences
    - General
    - UI Settings
    - Directories

- Connection
- Bandwidth
- BitTorrent
- Transfer Cap
- Queueing
- Scheduler
- Remote
- Playback
- Paired Devices
- Advanced
  - UI Extras
  - Disk Cache
  - Web UI
  - Run Program
- RSS Downloader
  - Favorites
  - History
- Miscellaneous
  - Add New Torrent
  - Add/Edit RSS Feed
  - Create New Torrent
  - Torrent Properties
    - General
    - Advanced
  - Tray Icon
- **Appendix B: Advanced Information**
  - Settings Directory
  - URL Formatting
  - Wildcards
  - Command Line Options
  - Keyboard Shortcuts
- **FAQ: Frequently Asked Questions**
  - Installation
  - Features
  - Usage
  - Network
  - Troubleshooting
  - Error Messages

- Incompatibilities
- Miscellaneous

- **Glossary**
- **External Links**
- **Changelog**
- **Credits**

« Previous    Next »

[« Previous](#)   [Next »](#)

µTorrent User Manual

# Chapter 01: Introduction to $\mu$ Torrent

This chapter provides a quick overview of  $\mu$ Torrent, presenting its main features and system requirements.

- [About  \$\mu\$ Torrent](#)
- [Features List](#)
- [System Requirements](#)

[« Previous](#)   [Next »](#)

[µTorrent User Manual > Chapter 01: Introduction to µTorrent](#)

# About µTorrent

µTorrent is a **BitTorrent client** originally created by Ludvig Strigeus (ludde), and is currently maintained by a team of developers at BitTorrent, Inc. A client is a computer program that follows the rules of a protocol. For example, HTTP (HyperText Transfer Protocol) is the protocol used for transferring web pages and other content, and your HTTP client (or web browser) is the program you use to get those web pages. Some popular browsers include Microsoft Internet Explorer, Mozilla Firefox, Opera, and Apple Safari. To an extent, they all work in the same way because they follow the same set of rules. Just as there are multiple HTTP clients, there are multiple BitTorrent clients that observe and conform to guidelines set in the BitTorrent protocol definition, and µTorrent is one such BitTorrent client.

So what sets µTorrent apart from other BitTorrent clients?

- **Micro-Sized Yet Feature Filled:** Most of the features present in other BitTorrent clients are present in µTorrent, including bandwidth prioritization, scheduling, RSS auto-downloading and Mainline DHT (compatible with BitComet). Additionally, µTorrent supports the Protocol Encryption joint specification and peer exchange.
- **Resource-Friendly:** µTorrent was written with efficiency in mind. Unlike many other BitTorrent clients, it does not hog valuable system resources, allowing you to use the computer as if it weren't there at all. Additionally, the program itself is very portable, being contained within a single executable small in file size.
- **Skinnable and Localized:** Various icon, toolbar graphic and status icon replacements are available, and creating your own is very simple. µTorrent also has support for localization, and with a language file present, will automatically switch to your system language. If your language isn't available, you can easily add your own, or edit other existing translations to improve them!

- **Actively Developed and Improved:** The developers put in a lot of time working on features and making things more user-friendly. Releases only come out when they're ready, with no schedule pressures, so the few bugs that appear are quickly addressed and fixed.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Chapter 01: Introduction to μTorrent](#)

# Features List

Although compact and resource-friendly,  $\mu$ Torrent does not skimp on features, with a feature set that rivals those of larger clients, such as Azureus, BitComet, and BitTornado.  $\mu$ Torrent's more notable features include:

- Bandwidth limiter
- Connectivity to Paired Devices
- Data transfer quota limiter
- Disk Cache system
- Distributed Hash Table (DHT) support
- Download bar
- Download scheduler
- Embedded tracker
- HTTPS tracker support
- Initial Seeding (Super Seeding)
- IP Blocklist support
- IPv6/Teredo support
- Local Peer Discovery (LPD)
- Local Tracker Discovery
- Localization
- Magnet URI support
- Micro Transport Protocol (uTP) support
- Multi-scrape support
- Multi-torrent interface with queueing support
- NAT Port Mapping Protocol (NAT-PMP) support
- NAT Traversal through UDP hole punching (uTP only)
- Peer Exchange
- Protocol Encryption
- Proxy support
- RSS "Broadcatching"
- Search bar
- Selective file downloading
- Streaming of media files

- UDP tracker support
- Unicode support
- Universal Plug and Play (UPnP) support
- Web Interface
- Web Seeding support

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Chapter 01: Introduction to μTorrent](#)

# System Requirements

µTorrent was designed with relatively basic system requirements in mind. µTorrent for Windows runs on Windows XP and newer.

The Mac client is compatible with Mac OS X 10.5 and newer. Mac users can download µTorrent [here](#).

Users of other operating systems [may be able to use µTorrent on their computers](#) with the use of a special piece of software that allows the operating system to run Windows applications. Note that there are no guarantees of any kind that everything will work properly when µTorrent is used on alternative operating systems.

[« Previous](#)   [Next »](#)

µTorrent User Manual

## Chapter 02: Basic Guides

This chapter provides basic guides for  $\mu$ Torrent. Be sure to read this chapter, especially if you are new to BitTorrent in general.

- [The Basics of BitTorrent](#)
- [Setup Guide](#)
- [Port Forwarding](#)
- [Downloading With  \$\mu\$ Torrent](#)
- [Downloading Using the RSS Downloader](#)

[« Previous](#)   [Next »](#)

[µTorrent User Manual > Chapter 02: Basic Guides](#)

# The Basics of BitTorrent

BitTorrent (often abbreviated as BT) is a peer-to-peer (P2P) **protocol** (a description and set of rules on how to do things) created by Bram Cohen, designed to distribute data in such a way that the original distributor would be able to decrease bandwidth usage while still being able to reach at least the same amount of people. Cohen's idea was to "break" the file being transferred into smaller segments called **pieces**. To save bandwidth, each person downloading (more commonly referred to as **peers** in the BitTorrent community) would have the pieces that they acquired available for upload to other peers in the **swarm** (the entire network of people connected to a single torrent). In this way, much of the load of sharing the file to every peer interested in it is offloaded to the peers. Note that a **seed** is basically a peer with every piece, so when a peer successfully attains all data in the torrent contents, that peer becomes a seed as well.

While on the surface, it appears that the only way to maintain a swarm's health is for there to always be a seed connected to the swarm, that is not the case. The most important factor to determining whether a swarm can continue to allow peers to complete a torrent is the **availability**. The availability of a torrent is the number of complete copies of the torrent contents there are distributed in the part of the swarm you're connected to, including the data you have. In most cases, if there is an availability of 1.0 or greater, then even if one single person does not have all the pieces, they are all still distributed across the entire swarm and can be acquired to form the complete file.

In order for everyone to be able to locate one another, there needs to be some centralized location that peers could connect to in order to obtain the other peers' IP addresses. BitTorrent **trackers** serve as this centralized location. In the most basic explanation, for each given swarm, a tracker only needs to collect a peer's IP address and port number to share with other peers connecting to that same swarm.

Because of the very nature of BitTorrent, speeds are not guaranteed

for any given torrent swarm. While you may get great speeds in one swarm, you might not in another. This is due to the fact that BitTorrent is a P2P protocol, so it depends on the upload speeds of the other peers you are connected to to generate your download speeds. A common misconception held by many people is that torrent swarms that contain more seeds and peers are faster than those with less. This is not always the case. There can be a swarm with only a few seeds and/or peers on fast Internet connections, and you'll be able to get great speeds from them, while a swarm with many more seeds and/or peers might contain mostly people with slow, dial-up Internet connections, will get you terrible speeds from them. In the same vein, *connecting* to more seeds and/or peers does not equate to greater speeds, and seeds don't necessarily give better speeds than normal peers.

[« Previous](#)   [Next »](#)

[µTorrent User Manual > Chapter 02: Basic Guides](#)

# Setup Guide

The focus of this section is to help you configure  $\mu$ Torrent to be able to obtain the optimal speeds for your Internet connection. While configuring it properly does not guarantee that you will hit your maximum upload and/or download speeds, it guarantees that  $\mu$ Torrent is doing the best it can to get good speeds. Be sure to read this entire section if you are unfamiliar with configuring  $\mu$ Torrent, because you will be expected to have read it already when asking for help elsewhere.

## The Setup Guide

When you open  $\mu$ Torrent for the first time, you are presented with the  $\mu$ Torrent Setup Guide. As stated in the wizard, following the simple directions will help you select the optimal settings for your Internet connection.

In the first part, you are asked to select your **upload** speed from the dropdown menu. If you do not know this information, you can test your Internet connection speed by selecting a location closest to where you are situated, and click the "*Run tests*" button at the bottom of the dialog. When you are running the speed test, make sure you are not using your Internet connection for anything besides the test. If you have any other computer on your network, disconnect them or shut them off before running the test. Run the test several times, and take the average of your upload speed given in the tests. After the speed test,  $\mu$ Torrent will automatically attempt to configure the settings based on the results of the test. If you wish, you can manually select your connection's upload rate from the dropdown menu, but do note the fact that there is a distinction between bits and bytes, and speed results are generally given in kbps (kilobits per second), which should not be confused with KiB/s (kibibits per second). Additionally, the tests may not be 100% accurate due to factors outside of your control, so if the closest option is only a little bit higher than what you received on the tests (perhaps by 10%), it's generally safe to select that. If the difference is greater than that, it's best

to select the lower option and [manually set the correct upload speed](#). Do not be tempted to select an option much higher than indicated on the speed tests in hopes that it will help you download faster, as it will not, and might end being detrimental to your speeds instead.

In the second part, a port is randomly selected for you the first time the Setup Guide is displayed, though you are free to change the port used. Alternatively, setting the port to 0 indicates to  $\mu$ Torrent that you would like for it to select a random port after the changes are confirmed. After you select your port, [left-click](#) "Run tests" to check that the port is open. It is essential that a port is open for  $\mu$ Torrent to listen for incoming connections on. If you are having trouble opening a port, continue reading onto [the port forwarding guide](#).

When you finish configuring everything, [left-click](#) "Save & Close" and you're done! If, for whatever reason, you need to return to this wizard to make a change, you'll find it accessible by selecting "Options" then "Setup Guide..." (or press [Ctrl](#) + [G](#)) in  $\mu$ Torrent. Alternatively, you can [left-click](#) on the [network status icon](#) in the status bar.

## Correcting Some Settings

If you manually chose a connection upload rate and found that you had to round down when selecting your connection type in the Setup Guide, you should adjust your upload speed limit to take advantage of the extra upload speed you actually have. Take the average upload speed you received when taking the test and divide it by 10, then round it to the closest whole number. Now use this calculated number as your [global maximum upload rate](#).

## Testing Your Configuration

Because torrents don't necessarily guarantee speeds, you can't just pick any random torrent and expect to be able to test the speeds properly. Luckily, there are many torrents out there that are seeded perpetually by computers sitting on fast broadband connections. Try a [test torrent](#) to test your configuration!

When properly configured,  $\mu$ Torrent should be able to attain the maximum speed possible for most common consumer Internet connections with these test torrents without much difficulty. Because these torrents are simply used for testing your connection speeds, it is safe to delete them whenever you are finished testing. Remember that these speeds are not indicative of how fast every torrent you come across will download.

## ISP Interference

Some Internet Service Providers (ISP) block or throttle BitTorrent connections because of the high bandwidth it generates due to the sheer number of people using BitTorrent. [Protocol Encryption](#) combats this attack vector by hiding the fact that connections are BitTorrent connections. Some ISPs cannot distinguish an encrypted connection from any other random data connection, so they are unable to label it as a BitTorrent connection, and consequently, cannot block or throttle it for being a BitTorrent connection. In general, there is no harm in enabling Protocol Encryption, other than a marginal increase in peer communication overhead.

Take note that some ISPs are starting to identify even encrypted BitTorrent connections with upgraded hardware, so even Protocol Encryption might not help users getting throttled by their ISPs. For a list of ISPs known to throttle, check the [Bad ISPs](#) list on VuzeWiki. If your ISP is known to throttle or block BitTorrent traffic, then you may want to consider setting the encryption to "*Forced*" and disable legacy incoming connections. If that fails, then the solution would probably be to switch to an ISP that does not throttle or block.

[« Previous](#)   [Next »](#)

[µTorrent User Manual > Chapter 02: Basic Guides](#)

# Port Forwarding

A large portion of the time, users complain that they are not getting great speeds with  $\mu$ Torrent. This can sometimes be a problem with an individual torrent (a lack of quality seeds, for example), but it is also possible that the user is not allowing incoming connections to reach his computer, and thus, is not making optimal use of  $\mu$ Torrent's ability to connect to peers with that torrent. Since  $\mu$ Torrent works best when it can connect freely to other peers, blocked or reduced access to other peers will adversely affect the download speeds a user can expect to achieve.

Being unable to accept incoming connections means your computer is in a **firewalled state**. In  $\mu$ Torrent, if you are unable to get a green **network status icon** after a long period of transferring different torrents, it is an indication that you might be in a firewalled state.

## Why Being Firewalled is Bad

Many firewalled users find themselves thinking "Hey, I'm firewalled, but I can still transfer the files, so it must mean I'm okay!" What they fail to realize is that being firewalled does not necessarily mean you cannot download at all, as firewalled users can still make outgoing connections, connecting to peers to transfer data in that way. While this is true, that's all you are limited to. That means that if someone else tries to initiate a connection with you, the attempt is blocked by your firewall.

You must then realize that you might not be the only firewalled user in the swarm. Since firewalled users can only make outgoing connections, and cannot accept incoming connections, it is natural that they (the firewalled users) cannot connect to each other. Being in a firewalled state not only cuts into the potential speed you could be attaining when not firewalled, but also means you are of very limited use to other peers in the swarm. Because there are less people available for firewalled users to connect to, they are open to less sources for data. Additionally, because they cannot accept incoming connections, other

peers do not connect to them, so they lose even more attention. Essentially, peers who are not in a firewalled state have the potential to connect to many more sources of data.

Removing yourself from being in a firewalled state does not mean you have to get rid of your firewall entirely. The only requirement is that you allow the application you wish to be unfirewalled in to listen to the port they want to through the firewall, also known as **port forwarding**. In the case of  $\mu$ Torrent, you need to set your firewall to allow  $\mu$ Torrent to listen to [the port set in the connection preferences](#).

Although it may sound like a security risk to "poke" a hole in your firewall, it is not the case. If no application is listening on the port that is opened in your firewall, any incoming connections on that port will be ignored. If an application is listening, security is up to that application. Unless there is a known, fully-remote exploit for the current version of  $\mu$ Torrent that would break your computer's security setup, there is no risk in opening a port on your firewall for  $\mu$ Torrent.

## Forwarding Ports in Your Software Personal Firewall

Nowadays, it is not uncommon for people to have a software personal firewall installed on their computers. While many people simply allow applications they recognize to access the Internet, oftentimes, it is not enough, as the firewall may continue to block the port that the allowed application is trying to listen on. As such, specific firewall rules may need to be created in order for  $\mu$ Torrent to work on a computer with a software personal firewall installed. The general rule of thumb you should follow is that you have to allow incoming TCP and UDP connections through the listening port set in the connection preferences. Because you are forwarding a specific port in your firewall, it is imperative that you do not have  $\mu$ Torrent [randomize the listening port each time it starts](#).

Because of the wide variety of software personal firewalls available today, there is no way to include specific instructions for every product.

Though this is the case, there is one specific firewall that  $\mu$ Torrent can create a firewall rule for automatically, and that is the Windows Firewall. **The relevant option** assumes you are running Windows XP with at least Service Pack 2 (SP2) installed or newer, and have the firewall enabled.

## Forwarding Ports in Your Router

With broadband becoming the prevalent way by which people access the Internet, and multiple computers in each home becoming a common sight, routers are often used to share the broadband connection across the computer network in the home. Even if multiple computers are not being used on the broadband connection, ISPs often supply routers for their customers to use. What many people don't realize is that routers themselves act like a firewall that, when left unconfigured, will generally leave your computer firewalled, even if you have your software firewall configured properly.

## Universal Plug and Play and NAT Port Mapping Protocol

As with software personal firewalls, there are a wide variety of routers available, and because of the sheer number, it is impossible to include port forwarding instructions for each router model in this user manual. Fortunately many routers support **Universal Plug and Play (UPnP)** or the **NAT Port Mapping Protocol (NAT-PMP)**, which allow  $\mu$ Torrent to open a port on the router automatically without user intervention, then close the port when the port is done being used. The problem with these zero-configuration protocols for automatically telling routers to forward ports is that they may not be supported by all routers, and different/incompatible implementations of the protocols may be included in many routers. By default, **UPnP** and **NAT-PMP** are enabled in  $\mu$ Torrent. If you find that you are still in a firewalled state, then it likely means your router does not support either protocol, or includes an implementation incompatible with the implementation used by  $\mu$ Torrent. If this is the case, then it is recommended that you disable these features in  $\mu$ Torrent, set up a static IP, and forward your ports manually.

## Setting Up a Static IP

On most routers, a connected computer's IP address on the network

is picked from a pool of IP addresses available for the router to choose from through **Dynamic Host Configuration Protocol (DHCP)**. The keyword here is "dynamic," as this indicates that each computer's IP address is assigned on-the-fly based on what IP addresses are still available in the router's pool of usable IP addresses. While some people get lucky and keep their LAN IP addresses for a long period of time, that is not a guarantee under DHCP. As such, port forwarding rules might work one day in forwarding traffic through a specific port to a specific computer at its LAN IP address at the time that the rule was made, but it may cease to work on another day because that specific computer's LAN IP address may have changed along the way. Some routers (notably, Linksys routers) don't even bother to forward ports to computers whose LAN IP addresses are within the DHCP IP range. Because of these reasons, it is necessary that you set up a **static IP address** for your computer (preferably, outside of the DHCP range) before you continue with forwarding your ports manually on the router.

Note that static IP refers to static LAN IP, which is different and unrelated to WAN IP. Your **LAN IP address** is the location of your computer within your network, but is not public for anyone to see besides the other computers within your network. Your **WAN IP address** is the IP address that people outside of your network see your network at, but it does not reveal the internal IP address allocation on your LAN, which may contain multiple computers, each (naturally) having their own LAN IPs. In the context of setting up a static IP for port forwarding, your WAN IP is irrelevant, and is normally not used in any step along the way.

The directions for setting up a static IP can be found on [PortForward.com](http://PortForward.com).

## Manual Port Forwarding

Assuming you have a static IP set up properly, the final step would be to do the actual port forwarding on your router. To get to the router configuration, you can normally visit the Default Gateway IP address in your web browser (you may have to append `http://` before the IP address in some web browsers for this to work). From there, you look for some method of forwarding ports or allowing/hosting "applications"

through the firewall, whereby you forward incoming connections on the listening port selected in  $\mu$ Torrent over both TCP and UDP to your computer's IP address, which should be the IP you selected when setting up the static IP. A list of many routers and port forwarding instructions for them can be found on [PortForward.com](http://PortForward.com). Be sure you know your router's exact brand and model. If your router is not listed there, you should consult with your router's documentation, or search the Internet for more detailed instructions.

## Testing Your Configuration

After you forward your ports, use the port checker from step 2 of the Setup Guide to test whether the port was opened correctly. If it confirms that the port is open, then you're done! If otherwise, then go over the previous instructions and make sure you did not skip a step or make any mistakes. If you're absolutely sure you configured everything properly, then there might be other problems at hand. In that case, you should read [the advanced guide on port forwarding](#).

[« Previous](#)   [Next »](#)

[µTorrent User Manual > Chapter 02: Basic Guides](#)

# Downloading With $\mu$ Torrent

Similar to needing a URL, like `http://www.utorrent.com`, to go to a website and download content, either a **.torrent file** or a **Magnet link** is required to download content available through BitTorrent. Most of the time, you can download this file from a website, though you can also get it from a friend or some other form of transfer. Many websites offer .torrent files as one method of downloading files available through that website. Sites that contain .torrent files are generally repositories of only the .torrent files, and usually don't create or directly make available any of the content being shared. These sites are either index sites or trackers. While **torrent index** sites list .torrent files for download, **torrent trackers** merely coordinate the swarm. Many torrent trackers function as a torrent index as well, listing the torrents that it tracks (and sometimes, torrents from external trackers as well).

So where do you go about looking for these .torrent files? Searching with your favorite search engine, and attaching the word *torrent* to the query generally works wonders in finding you decent results, but  $\mu$ Torrent also includes a **built-in search bar** to some of the more popular .torrent file search engines.

## Adding a Torrent

Once you obtain the .torrent file you wish to download, you simply import it into  $\mu$ Torrent. There are several ways of achieving this in  $\mu$ Torrent:

- **Double-click** the .torrent file or the Magnet link (only if .torrent files are **associated with  $\mu$ Torrent**)
- **Drag-and-drop** the .torrent file into  $\mu$ Torrent
- Select "*File*" then "*Add Torrent*" (or press **Ctrl + O**) in  $\mu$ Torrent and open the .torrent file
- If you know the direct URL to the .torrent file, but don't have it on your hard drive, you can select "*File*" then "*Add Torrent from URL*"

(or press **Ctrl** + **U**) in  $\mu$ Torrent and enter the URL of the .torrent file

After opening the .torrent file, tell  $\mu$ Torrent where you'd like the torrent contents to be saved. If  $\mu$ Torrent doesn't automatically start downloading, you can start the torrent job manually by selecting the torrent job in the list and pressing the "Start" button on the  $\mu$ Torrent toolbar, or by right-clicking the torrent and selecting "Start". If you are using a Magnet link, you will need to wait a few seconds for the Meta-data to be retrieved.

## What to do After the Torrent Job Finishes Downloading

After a torrent job finishes downloading, you may view the files that you have downloaded. While you can also remove the torrent job from the torrent jobs list if you so wish, you are highly encouraged to leave the torrent job **seeding** (uploading after you have obtained every piece). Although the length of time that you should leave the it seeding is not defined in particular, it is recommended that you share until the amount of data you upload reaches at least the same as the amount of data that you have download, also known as reaching a 1.0 **ratio**. This ratio is calculated by dividing the amount of data you have uploaded by the amount that you have downloaded. Granted, it is technically impossible for every person in any given swarm to reach a 1.0 ratio, but people who leave the swarm before even getting close to that ratio are abhorred in the BitTorrent community, and are labeled as **leechers** (which carries a strong negative connotation). Because leechers have a detrimental effect on swarms, some people resort to vigilante tactics and block connections to leechers. Additionally, private trackers may ban leechers.

[« Previous](#)   [Next »](#)

[µTorrent User Manual > Chapter 02: Basic Guides](#)

# Downloading Using the RSS Downloader

RSS is a function in  $\mu$ Torrent to automatically view and download torrents from your favorite torrent site! Sites like [DailyTvTorrents](#) publish their torrents using RSS feeds, allowing for clients such as  $\mu$ Torrent to display new torrents within them.

## Adding a feed

Click on the Add RSS icon in the toolbar, or right click on All Feeds and choose "Add RSS Feed." Enter your RSS feed's URL into the input box, and check Custom Alias if you want to give a name to the feed.

## Disabling a feed

To disable a feed, simply right click the feed in the category list and choose "Disable Feed." If you want to re-enable it, simply right click again and choose "Enable Feed."

## Renaming a feed

You can rename a feed or edit the URL by either right clicking it and choosing "Edit Feed" or double clicking on it.

## Viewing releases on an RSS feed

To view the releases for an RSS feed, click on the feed. Alternatively, click on "All Feeds." You can also view multiple feeds at once by holding Ctrl and clicking multiple feeds, or even combine releases and your torrents by clicking on a feed and a label or category for loaded torrents. All the releases currently on the feed will show in the main listview. Double clicking will attempt to download the .torrent for the release. To open the release's URL in your browser, right click it and

choose Open URL in Browser.

Extra columns of information are available for RSS feeds. To enable or disable them, right click on the name of any column. You can also click and drag columns to rearrange them. These settings are independent of the columns available when viewing your loaded torrents.

## Using feeds that require HTTP authentication

For feeds that require HTTP authentication, simply use this format for the feed URL:  
`http://username:password@sometorrentsite.com/rss.php`

## Using feeds that require cookies

To use feeds that require cookies, you must find the cookie for the site, and grab UID and pass from it.

- IE users will find their cookies in %UserProfile%\Cookies
- Firefox users will find their cookies in Tools -> Options -> Privacy -> Show Cookies
- Opera users will find their cookies in Tools -> Advanced -> Cookies
- Users of other browsers will have to consult their browser's documentation

Once you have the appropriate information, use this format for the feed URL:

`http://sometorrentsite.com/rss.php:COOKIE:uid=1234;pass=asdjh123`

Not every site uses uid and pass as the cookie variables, or use additional ones, so one MUST use the exact variable name and the extra variables they specify. For example, a certain site uses id, password, and secure as its cookie variables.

## Automatically downloading torrents

To automatically download all torrents from a feed, you can open the

Edit Feed dialog and choose "Automatically download items published in feed." Checking the smart episode filter will make it only download the first version of a particular episode. The smart episode filter not work if the "Episode" column is blank.

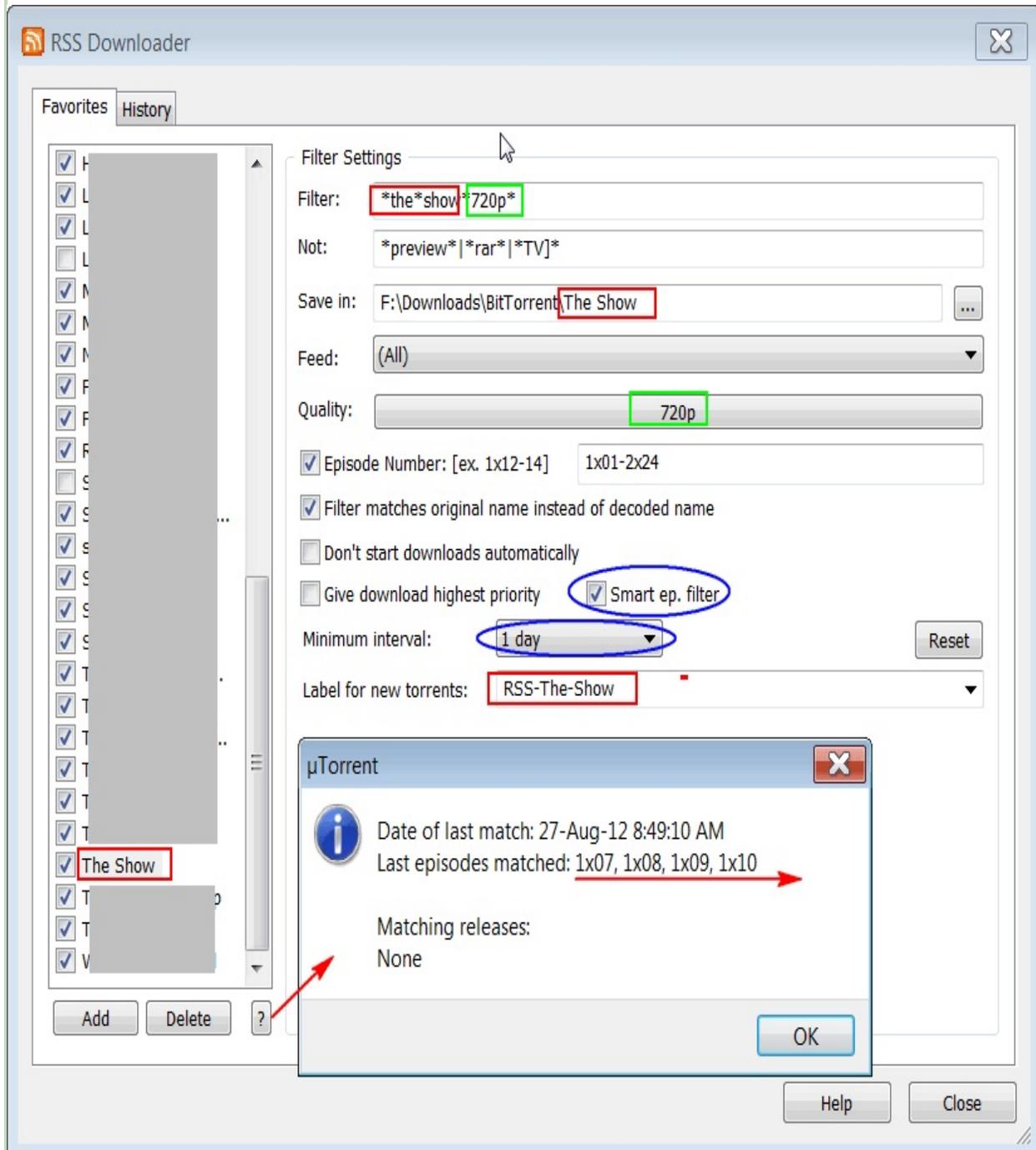
If you want to download only specific releases, there are two ways to do that. The simpler method is to right click on the relevant release in the list and choose "Add to favorites." This will add it to your Favorites list and bring up the RSS Downloader dialog for further editing of your favorite.

The alternative method to automatically download specific releases is to right click on any feed or "All Feeds" and choose "RSS Downloader," or press Ctrl-R. This will bring up the "RSS Downloader" dialog.

- Hit Add, then type the name of the favorite.
- Click on the editbox next to Filter: and add your filter. The allowed wildcards are \* and ?
  - Multiple filters can be entered by separating each one with a |
  - An example of a filter you could use would be \*MP3 Archives\*
  - If you have a strange feed using underscores or something besides periods, you can also try something like \*Go?Open\*
  - Do not specify an episode or season number here. This will break the filter. Use the episode number function instead. Adding them here will only work if "filter matches original name instead of decoded name" is checked. This may be required for non-standard episode formats or dates.

This is the minimum required to setup a favorite. The rest

of the functions are extra features for more control of the automatic downloads. The following example setting will download all episodes of "The Show" with a 720p resolution:



- Not: allows you to exclude certain strings from matching. An example is you don't want releases with AC3 audio and H.264: you

can write `*AC3*|*H*264|*x*264*` in Not: to exclude those.

- By default, torrents will automatically download to My Documents\Downloads. If you want the torrents to download to a specific folder, you must either specify a folder in Save in: or set a default download path in the "Directories" section of µTorrent's preferences.
- Feed: chooses what feed you want the filter to apply to; either all of them or a specific one
- Quality allows you choose various qualities to match against, or allow all. You can choose more than one quality in the dropdown list.
- Episode number is to download only specific ep numbers, say to avoid releases of old episodes. It supports multiple formats. For example, to download only the first 12 episodes of season 1, type in 1x1-1x12 or 1x1-12. Or to download starting from episode 13 of season 1 and include all later seasons, type in 1x13-
- "Don't start downloads automatically" will add any downloaded torrents in Stopped mode. They will be added to your list, but will not begin downloading until you choose them from the torrent list and start them.
- "Filter matches original name instead of decoded name" is so that you can match based off what the original name is in the feed (given in the Name column), instead of µTorrent's parsed result.
- Give download highest priority sets all torrents downloaded automatically through RSS to the top of the queue, making your seeds and other downloads get queued if you reach the max active torrents.
- Smart ep. filter makes µTorrent only download the first version of each new episode that matches your filter. This function will only work if the "Episode" column is not blank.
- Minimum interval sets a minimum interval between matches for the filter: if you set it to 2 days, µTorrent will not download anything for at least 2 days after a match.
- You can use the Reset button to make µTorrent forget that it has downloaded episodes and the last time matched for that filter.
- Label for new torrents auto-sets a label for torrents that match the filter.
- All changes are automatically saved when you make them, so you

don't have to do anything once you're done editing the filter.

- You can select a filter and press ? to see what the last four episodes were that matched, and the last time that the filter matched something. A list of the currently matching episodes from the feeds you've defined for it is also available in the "?" dialog, so you can double-check your filter expression.

## Sorting favorites

You can sort your favorites by by dragging and dropping them to wherever in the list you'd like them.

[« Previous](#)   [Next »](#)

µTorrent User Manual

## Chapter 03: Advanced Guides

This chapter is designed to explain more advanced aspects of  $\mu$ Torrent. For the most part, users will not encounter the following within the scope of normal use.

- [More Port Forwarding](#)
- [Migrating to Another Location](#)
- [Switching from Another Client](#)
- [Alternative Operating Systems](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Chapter 03: Advanced Guides](#)

## More Port Forwarding

Even after having carefully and laboriously followed all the [port forwarding instructions previously written](#), you've found that your port simply refuses to open up. So what exactly is the problem? Possibilities include (but are not limited to):

- **Intentional ISP interference:** Some ISPs are known to simply block connections to unauthorized ports, in which case your entire network is essentially placed into a firewalled state.
- **Internet connection type:** One prominent example of this being the source of the problem are Internet connections that are received wirelessly (though satellite or something similar). Customers of wireless connections are very often not given WAN IP addresses, and so are permanently in a firewalled state. In fact, any kind of connection where you are not in control of the NAT will generally leave you in an unfirewalled state. Such is the case with university-provided connections, where network administrators generally block connections to unauthorized ports.
- **Proxy service:** Some ISPs place their users behind a transparent proxy, whereby the port checker might be unable to detect the forwarding state properly. In that case, try a [test torrent](#) and let it run for a while. If the [network status light](#) turns green, then everything's probably configured properly. Even then, though, being behind a proxy essentially places one behind a firewall, so you might still suffer problems as if you were behind a firewall.
- **Network hardware blocking:** Some modems are known to cause issues with your computer being in a firewalled state even though they technically aren't routers. One notorious example is the Motorola SurfBoard brand of modems, which you can read up about [at the PortForward.com forums](#).

User error is also a very common problem when it comes to port

forwarding issues, but assuming that everything in the [basic port forwarding guide](#) was followed carefully, and none of the above possibilities are applicable, then the problem very likely lies with another issue called **double NAT**. Unlike any of the other issues listed above, double NAT problems can often be taken care of, provided the user follows the necessary steps as described below.

## Double NAT

Double NAT occurs when your computer is sitting behind two or more routers. In most double NAT cases, it turns out that the user has a dedicated router, but was also unknowingly provided with a modem by their ISP that came with router or firewall capabilities. When that is the case, the user simply forwards ports from the router to the computer, leaving the modem alone, and this is exactly where the problem lies. Because the modem acts as a router as well, if it is not configured properly, it essentially means that the user remains in a firewalled state, since the dedicated router that the user did configure is not actually receiving incoming connections on the forwarded port due to everything being blocked by the outermost router -- the modem. Be aware that this is a specific case of the issue at hand. In more severe cases, users can have more than just two routing devices, and rectifying the problem can become that much more difficult, depending on the solution taken.

## Removing or Disabling the Extraneous Router

In the simplest of cases, ridding yourself of the double NAT situation comes down to simply removing the extra routers, or disabling their routing capabilities. Using this method assumes that the extraneous routers being operated upon are absolutely unneeded on the network. If that is not the case, then the only solution you have is to [chain port forward](#). With the method being described in this section, only one router should end up on the network, that router being the one that your computers are actually connected to. Some examples are as follows:

- You have a modem that acts as a router, a dedicated router that is connected to the modem, and then your computers that are connected to the router. In this case, you should disable the routing

capability in the modem, so that you're left with only the dedicated router as the sole router on the network (which is the one connected to your computers).

- You have a modem that acts as a router, two routers, and all of the computers connected to the same router. In this case, you should disable the routing capability on the modem, and remove the router that no other computers are connected to.

As you can see, the general rule of thumb is that you remove all extraneous routers. You'll notice, though, that the modem with routing capability never gets removed -- that's because the modem function is important for allowing you to actually connect to the Internet. In any case, to disable the router in the modem, you have to physically connect a computer directly to the modem, then visit the configuration page for the modem. Before actually disabling the routing capability, you must be sure to check whether your modem contains login information for your ISP. If it does, then you are going to have to make sure you have a copy of that information on hand. This is most often the case for people using a DSL modem and router, where login information is usually stored where the PPPoA/PPPoE configuration page is. That aside, setting the modem to bridge mode is what you should be combing through its configuration page for. After you do this, everything should hopefully be fine. If you find that your Internet connection no longer works, you should fill the login information you copied into the appropriate location in the remaining active router on your network (if you copied PPPoA/PPPoE settings from the modem, copy it into the PPPoA/PPPoE settings in the router).

## Chain Port Forwarding

This method can be very annoying, as it requires that you set static IPs for and forward the desired port through each and every one of them. Basically, you need to follow [the basic port forwarding guide](#) for each and every router, except that the IP you're forwarding to is the IP of the next router in the chain of routers leading up to your computer. Each router must be assigned a static IP address, which can normally be set in its configuration pages. While specifics can't be delved into because of the

sheer number of different routers available, the following case example might be of use to illustrate the process more clearly:

- You have a modem that acts as a router. A dedicated router (router A) is connected to it. Another dedicated router (router B) is connected to router A. Your computer is connected to router B.
- The modem's routing subnet starts with 192.168.1.x. It forwards the port specified in  $\mu$ Torrent to router A, which is at IP address 192.168.1.5 on the modem's subnet.
- Router A has a static IP set to 192.168.1.5. Its own subnet starts with 172.16.1.x. It forwards the port specified in  $\mu$ Torrent to router B, which is at IP address 172.16.1.3 on router A's subnet.
- Router B has a static IP address set to 172.16.1.3. Its own subnet starts with 10.0.0.x. It forwards the port specified in  $\mu$ Torrent to your computer, which is at IP address 10.0.0.6 on router B's subnet.
- Your computer has a static IP address set to 10.0.0.6, and because the port was forwarded from the modem to router A, and from router A to router B, then from router B to this computer, the port checker considers your client to be connectable.

Please be aware that that was just an example. Many conditions, including the IP addresses, the number of devices on your network setup, or how everything is connected, will most likely differ from the example. All you can do is adapt the example to your situation and configure everything accordingly. Setting the static IP addresses up properly is extremely important when chain port forwarding. Any mistakes (or failure to do so) means more troubleshooting in trying to figure out which device's IP address changed if your port suddenly becomes unforwarded.

## More Troubles

If you do not see an answer or solution above, or you are unsure of what your problem is, please join [the IRC channel](#), or [the forums](#) to ask for more assistance. Do explain what you've tried so far with some level of detail so that people know what you have attempted, and what else you could try (if anything).

[« Previous](#)   [Next »](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Chapter 03: Advanced Guides](#)

## Migrating to Another Location

In some cases, you might find that you need to move your files, be it due to simple reorganization, possibly to reinstall Windows, or maybe even to move  $\mu$ Torrent and all the torrent jobs along with their contents to a new computer. In any case, the process is simple, though it can turn out to be a lengthy one. Realize that along the way, you might lose your accumulated statistics for each torrent job, but that does not mean you lose the statistics on the associated trackers, so do not fret about that.

## Reinstalling Your Operating System

In the simplest of cases, reinstalling your operating system only requires you to make a backup of the data onto a new drive or partition if the torrent contents and  $\mu$ Torrent settings directory are located on the same drive or partition as the operating system you are trying to reinstall. After everything is completed, you simply have to move the files back to their previous locations, and  $\mu$ Torrent will resume everything without issue. If you did not have an encapsulated "installation" of  $\mu$ Torrent, then make sure you have the same username before moving the settings directory back into %AppData%. If you are unable to perform any of the above, you will have to take the long route of performing the migration as if you were really moving the torrent contents.

## Moving to a New Computer

Migrating  $\mu$ Torrent to a new computer simply requires that you copy your files to your new hard drive, along with backing up the entire  $\mu$ Torrent settings directory. The most difficult part of this process comes with the placement of the  $\mu$ Torrent directory and the torrent contents. If you want to complete the process with minimal effort, it becomes a simple matter of making sure the paths for all the files related to  $\mu$ Torrent and the torrent contents on the new computer are identical to the respective paths on the old computer.

- If you had an encapsulated "installation" of  $\mu$ Torrent, an identical username is not necessary. Otherwise, if the  $\mu$ Torrent settings directory was located in %AppData%\uTorrent, then you will have to create a user with the same exact account name on the new computer as on the old computer, then move the  $\mu$ Torrent settings directory into the proper location on disk.
- If you plan on moving the drive over to the new computer, then as long as the drive letter remains the same, you save yourself the headache of dealing with paths. Otherwise, if you're going to copy the files from the old drive to the new drive, then you should make sure the torrent contents' paths remain identical.

If any of those tips fail, and you are unable to keep paths identical for either the settings directory or the torrent contents, then you're in for a very long ride, and will have to perform everything as if you had moved the torrent contents.

## Moving Torrent Contents

This process, if you haven't already figured it out, might require the most amount of time out of any of the "migration" processes, as it might require that you allow  $\mu$ Torrent to verify the data integrity of the torrent contents after the move. What's time consuming isn't really the procedure itself, but the fact that rechecking can take a lot of time, depending on the torrent contents' sizes, and how many different torrent jobs you want to move. All that's needed when you're moving torrent contents is that you stop the torrent job in  $\mu$ Torrent, move the torrent contents to wherever you need them to be, [set the download location](#) for each associated torrent job for the moved contents, and start the torrent job. If  $\mu$ Torrent doesn't recognize the existing data, stop the torrent job and [force re-check](#) for each relevant torrent job.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Chapter 03: Advanced Guides](#)

## Switching from Another Client

Now that you've made the switch to  $\mu$ Torrent, you're probably wondering if or how you could import all the torrent jobs you were running in your previous client. Rest assured that the process is a relatively simple one, and only requires a bit of patience, depending on how much data is being "imported" into  $\mu$ Torrent -- the larger and more numerous the torrent contents being imported, the longer it will take, because it all means that the data integrity checking will take longer as well.

### Removing Extra Extensions

There is a simple matter to tend to first before proceeding with the import, and that has to do with a specific feature in some other clients. Some other clients append a special extension to incomplete files to indicate that they are incomplete, and  $\mu$ Torrent (if [the relevant option](#) is enabled) is no exception. Because of this extension,  $\mu$ Torrent might have trouble recognizing the torrent contents, and will ignore them instead. To fix it, all you have to do is remove the special extension from any incomplete file. If there are many files, this can quickly become a boring and tedious task. Luckily, a simple batch script can be used to rename all files with the unwanted extension:

```
@for /r %%i in (*.XT!) do @move "%%~fi" "%%~dpni"
```

Copy and paste the above line of code to a new text file, then change the XT! to the extension you wish to remove. BitComet, for example, appends a bc! extension to the end of incomplete files, so you would replace the XT! in the script with bc!. Now, save the text as a batch file (rename.bat, for example), and all you have to do is move the batch file to the directory containing files you want to rename, and run the script, then repeat for any other directory containing files you want to remove the specified extension for. Note that the above script is recursive, so any files contained within any other directory in the same

directory as the batch file will be renamed as well if they have the specified extension.

## Importing Torrent Jobs

With extra extensions out of the way, the rest of the process is smooth sailing in that you'll likely get no errors. All you need to do is to open the .torrent file in  $\mu$ Torrent, and point it to the location that the torrent contents were already saved to while you were using your other BitTorrent client.  $\mu$ Torrent will automatically check the data integrity, and assuming everything went well, will resume where you left off previously. Realize that you must do this for each and every .torrent file you wish to import.

There is an alternative method for loading .torrent files that does not require manual user intervention, but it's best left for people with many torrent jobs to import. Additionally, there is one major prerequisite before this method can be used: all torrent contents you wish to import must be located in a common directory, not organized in any fashion, but simply located in that directory as if it were the directory you selected to download the torrent contents to. That aside, you can start setting the process up as follows:

1. In the **Directories preferences**, tick the *"Put new downloads in"* checkbox, untick the *"Always show dialog on manual add"* checkbox, and select the path to the directory that all your torrent contents are located in. Make sure the *"Don't start the download automatically"* option is left unticked.
2. Now create a new directory somewhere, then go to the **auto-load torrents option** and set it to point to that newly-created directory.
3. After confirming the changes, move all the .torrent files you wish to load into the directory you just told  $\mu$ Torrent to automatically load .torrent files from, and  $\mu$ Torrent should do so, automatically starting and checking all the torrent contents.

If a torrent job is already completed,  $\mu$ Torrent will automatically place it in seeding mode. After everything is imported, feel free to revert the

changes made to the  $\mu$ Torrent settings.

## Changing .torrent Associations

Another thing you might want to do when switching to  $\mu$ Torrent is to unassociate .torrent files with your previous client, and have them open with  $\mu$ Torrent by default. If you plan on keeping the other client, and want  $\mu$ Torrent to open .torrent files by default, you should look in your previous client's options and find a way to have it not associate itself with .torrent files automatically, if it does so. In any case, all you have to do in  $\mu$ Torrent is **associate it with .torrent files** in the Preferences.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Chapter 03: Advanced Guides](#)

# Alternative Operating Systems

In a world where most end users only know about or are familiar with Microsoft Windows, it's easy to forget that there are alternative operating systems out there, right? For those of you who aren't using Windows, it's still possible to run  $\mu$ Torrent in your operating system, it just takes a bit more fussing around to get it running more nicely (though there might still be residual problems in functionality due to operating system limitations).

## Running on Wine (\*BSD, GNU/Linux)

Wine is an open source project that aims to implement the Windows API on Unix-like operating systems. In simpler terms, it is an attempt to allow users running a Unix-like operating system (such as \*BSD or GNU/Linux) to execute applications that normally run only on Windows. The first step in running  $\mu$ Torrent on Wine would, naturally, be to download and install the latest version of Wine from [WineHQ.org](http://WineHQ.org) if you haven't already done so.

When Wine is installed, the only thing you need to do is run  $\mu$ Torrent by using the `wine` command in a terminal window. Wherever you have your  $\mu$ Torrent executable located at on your filesystem, you simply run  $\mu$ Torrent with the command `wine FULL_PATH` (where `FULL_PATH` is the full path to the  $\mu$ Torrent executable) in in the terminal window. Naturally, there are many different (and easier) ways to execute  $\mu$ Torrent through Wine, but because there are infinitely many combinations of softwares available for use on the Unix-like operating systems, the most generic way will have to suffice.

So where exactly is the hard part in using  $\mu$ Torrent on Wine? It's in the limitations that come with using Wine and its incomplete implementation of the Windows API. Because of this, and the fact that Wine is unable to integrate too deeply with its host operating system, there are several known issues you should be aware of:

- **Problem workarounds:** Some known issues with Wine have been worked around in  $\mu$ Torrent with a set of "hacks" enabled through [sys.enable\\_wine\\_hacks](#).
- **Boss-Key:** [Boss-keys](#) do not work under Wine.
- **Graphical oddities:** There are several oddities in the interface running  $\mu$ Torrent under Wine...
  - By default, the interface sticks out like a sore thumb, with the gray Windows application window background in all its glory. Users can modify the theme colors in the "*Desktop Integration*" tab of `winecfg` to better integrate Wine-run applications with surrounding applications.
  - If the list-view has a black background, then make sure the Windows version is set to Windows XP in `winecfg`.
- **Open Containing Folder:** Because [Open Containing Folder](#) attempts to open a Windows Explorer process, and the native Windows Explorer is not present on alternative operating systems, it will open the Wine implementation, which looks unimpressive, and does not work very well. Unless you can tolerate it, don't bother using this feature under Wine.
- **Open File:** "*Open*" in [the torrent jobs list context menu](#) or on [the Files tab](#) will not work, as they attempt to run a Windows application associated with the file type, but since there are none, they fail.
- **Settings directory:** Because Windows environment variables do not apply in alternative operating systems, it naturally follows that `%AppData%` does not either. You'll find that the  [\$\mu\$ Torrent settings directory](#) is actually located at `~/.wine/drive_c/windows/profiles/USERNAME/Application Data/uTorrent`, where USERNAME is your username on the operating system.
- **Start on system startup:** Because alternative operating systems have different methods of booting applications at startup, the usual methods of adding a startup entry in Windows (be it in the Registry, or in the Start menu) won't work. [Start  \$\mu\$ Torrent on system startup](#)

should be disabled.

- **System tray:** The [system tray feature](#) has been reported to be flaky under Wine, so if you have issues using it, you should disable all relevant options.
- **.torrent file association:** Due to the inability of Wine to map file associations to applications run through Wine, the feature is rendered useless. [Check association on startup](#) should be disabled. An alternative method of "emulating" file association would be to use the [auto-load torrents](#) feature to tell µTorrent to automatically load files from a directory on your filesystem. In that way, you can simply drag or download .torrent files into that directory, and if µTorrent is running, it will automatically load the new files.

## Running on Darwin (Mac OS X)

Note: There is a native µTorrent client for Mac available [here](#). If you have an older version of Mac OS X (10.4 or below) you may need to use the instructions below.

[Darwine](#) is a port of Wine for use on Darwin and Darwin-based operating systems, such as Apple's Mac OS X. Darwin requires an X server to work, so if you don't already have it installed, you can find the installer in the Apple Restore DVD, or by downloading it along with Apple's [XCode](#).

Installing Darwin should be a cinch. Just as with Wine, it's generally best to download and install the latest version of Darwin. After installed, you should be able to simply double-click µTorrent's executable file, and Darwin should automatically run the application. If not, then run `WineHelper.app` (which should be located in the installed Darwin directory) then open µTorrent's executable file. That's all!

Not unlike Wine, there are several known issues in using µTorrent with Darwin that you should be aware of:

- **Shared bugs:** Due to its heritage, Darwin exhibits [all of the same](#)

issues that Wine does.

- **Drag-and-drop:** Drag-and-drop doesn't work under Darwin.
- **Secondary-click:** **Option** + **left-click** doesn't send a **secondary-click** event to  $\mu$ Torrent because it has not been implemented in Wine.

[« Previous](#)   [Next »](#)

µTorrent User Manual

# Appendix A: The $\mu$ Torrent Interface

This appendix provides detailed descriptions of most of the interface elements in  $\mu$ Torrent.

- **Main Window**
  - Main Menu
  - Toolbar
  - Category List
  - Torrent Jobs List
  - Detailed Info Pane
    - General
    - Trackers
    - Peers
    - Pieces
    - Files
    - Speed
    - Logger
    - Related
  - Status Bar
- **Preferences**
  - General
  - UI Settings
  - Directories
  - Connection
  - Bandwidth
  - BitTorrent
  - Transfer Cap
  - Queueing
  - Scheduler
  - Remote
  - Playback
  - Paired Devices
  - Advanced
    - UI Extras
    - Disk Cache

- Web UI
- Run Program
- **RSS Downloader**
  - Favorites
  - History
- **Miscellaneous**
  - Add New Torrent
  - Add/Edit RSS Feed
  - Create New Torrent
  - Torrent Properties
    - General
    - Advanced
  - Tray Icon

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface](#)

# Main Window

The main window is the part of the  $\mu$ Torrent interface that you'll likely be interacting with the most, so it's good to get familiar with it by reading this section of the help.

Several sections of the main window have column-based, tabular information. Note that **left-click**ing on a column will sort the items in the list by that column, alternating between ascending and descending sort for each additional **left-click**. You can select what columns you wish to see by **right-click**ing the column headers. If you wish to reset all changes you've made to the columns, you can select "Reset" in that context menu.

- **Main Menu**
- **Toolbar**
- **Category List**
- **Torrent Jobs List**
- **Detailed Info Pane**
  - General
  - Trackers
  - Peers
  - Pieces
  - Files
  - Speed
  - Logger
  - Ratings
  - Related
- **Status Bar**

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Main Window](#)

# Main Menus

The main menus are where you'll find many of the other facilities included in  $\mu$ Torrent, such as the [Preferences](#), or the [RSS Downloader](#).

## File

- **Add Torrent...** [Ctrl](#) + [O](#) prompts you for the location of the .torrent file that you are trying to open on disk. If you set a [default download location](#), the torrent job will automatically be added to the torrent jobs list and started (if you did not set  $\mu$ Torrent to [start torrent jobs in stopped mode](#)). If you did not set a default download location, you will be prompted where you would like to save the torrent contents, with the [Add New Torrent](#) dialog being shown by default.
  - **Add Torrent (no default save)...** [Ctrl](#) + [D](#) does the same thing as [Add Torrent](#), but will always ask you where you'd like to save the torrent contents, regardless of whether you have a [default download location](#) set or not.
  - **Add Torrent from URL...** [Ctrl](#) + [U](#) allows you to open a .torrent file directly from URL without having saved it to your hard disk. Depending on the method used by the website for authenticating users (if any), the URL may have to be entered [in a specific format](#). Magnet URIs may be used here, and  $\mu$ Torrent will attempt to download the info dictionary from the corresponding .torrent file from any other client that also supports the URI scheme.
  - **Add RSS Feed...** opens up the [Add RSS Feed](#) dialog, where you can add RSS feeds for  $\mu$ Torrent to monitor.
  - **Add Device...** replaces the main view with a [Devices](#) pane, so that you can add a new remote paired-device to drag torrents onto it.
- 
- **Create New Torrent...** [Ctrl](#) + [N](#) lets you [create .torrent files](#) that you

can use to share your data with other people.

---

- **Exit** does just that -- it exits  $\mu$ Torrent. Note that exiting  $\mu$ Torrent automatically sends a stop signal to trackers, so you do not have to stop all torrent jobs before exiting. Torrent jobs that are not stopped will automatically be started when  $\mu$ Torrent is next opened. Even after exiting, the  $\mu$ Torrent process might continue to run for a while longer. This happens because  $\mu$ Torrent is trying to finish transferring pieces and write remaining pieces to disk from memory. The process should disappear after a few seconds.

## Options

- **Get  $\mu$ Torrent Plus** opens up an info-pane with  $\mu$ Torrent Plus information and related links.
  - **Preferences** **Ctrl** + **P** opens up the **Preferences** dialog, where you can configure most of  $\mu$ Torrent's options.
  - **RSS Downloader** **Ctrl** + **R** opens up the **RSS Downloader** dialog, where you can configure  $\mu$ Torrent's RSS settings.
  - **Setup Guide** **Ctrl** + **G** opens up the **Setup Guide** dialog, where you can tell  $\mu$ Torrent to configure some basic settings that can affect your speeds.
- 
- **Show Toolbar** **F4** toggles the display of the **toolbar** at the top of the main window.
  - **Show Detailed Info** **F5** toggles the display of the **information pane** near the bottom of the main window.
  - **Show Status Bar** **F6** toggles the display of the **status bar** at the bottom of the main window.
  - **Show Category List** **F7** toggles the display of the **category list** at the left of the main window.

- **Show Featured Content** **F8** toggles the display of the 'Featured Content' category-node at the left side category pane.
  - **Show Devices** toggles the display of the 'Devices' category-node at the left side category pane.
  - **Show Plus Information** toggles the display of the 'µTorrent Plus' offer-link at the bottom of the left side category pane.
  - **Narrow Toolbar** **F11** toggles the width of the **Toolbar** at the top of the main window. This also hides the *Add RSS Feed* icon when enabled.
  - **Compact Category List** **F12** toggles the display of dividers between sections at the left of the main window.
- 

- **Icons on Tabs** toggles the display of the graphic icons on the tabs at the top of the **information pane**.
- 

- **Auto Shutdown**

- **Disabled** tells µTorrent not to perform any of the following actions. Note that besides this option, all options in this submenu get disabled after they have occurred.
- **Quit when Downloads Complete** exits µTorrent when all currently downloading torrent jobs reach 100% download completion.
- **Quit when Everything Completes** exits µTorrent when all currently active torrent jobs are completed.
- **Hibernate when Downloads Complete** puts the computer into hibernation mode when all currently downloading torrent jobs reach 100% download completion.
- **Hibernate when Everything Completes** puts the computer into hibernation mode when all currently active torrent jobs are completed.
- **Standby when Downloads Complete** puts the computer into standby mode when all currently downloading torrent jobs reach 100% download completion.
- **Standby when Everything Completes** puts the computer into standby mode when all currently active torrent jobs are completed.
- **Reboot when Downloads Complete** restarts the computer when all currently downloading torrent jobs reach 100% download completion.

- **Reboot when Everything Completes** restarts the computer when all currently active torrent jobs are completed.
- **Shutdown when Downloads Complete** shuts the computer down when all currently downloading torrent jobs reach 100% download completion.
- **Shutdown when Everything Completes** shuts the computer down when all currently active torrent jobs are completed.

## Help

- **Learn more about µTorrent Plus** opens an info page in the main view with µTorrent Plus related information.
- **µTorrent Help** **F1** opens the user manual.
- **µTorrent FAQ** opens the **FAQ**.
- **µTorrent Webpage** opens the **µTorrent webpage** in your default web browser.
- **µTorrent Forums** opens the **µTorrent forums** in your default web browser.
- **Send Beta Feedback** opens an email for sending user comments and issues related to the current beta release being developed.
- **Check for Updates** manually tells µTorrent to check for a newer stable build from the µTorrent servers. µTorrent will continue to check for newer versions once every 24 hours after this is selected if you have enabled the option to **check for updates automatically**.
- **Download Translation** tells µTorrent to download the latest translation file and place it in the **settings directory**.
- **Show Statistics** opens a dialog displaying basic statistics that

µTorrent has collected.

---

- **About µTorrent** opens the About dialog, which displays the credits, version, and build number, among other things...

« Previous    Next »

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Main Window](#)

# Toolbar

The buttons on the toolbar at the top of the  $\mu$ Torrent interface allows you perform some basic functions, most of which apply to the currently selected torrent(s). The order of the list here reflects the order of the buttons in the toolbar. If you are unsure, you can place your mouse cursor over a button, and a tooltip should pop up providing the description of the button. The toolbar's visibility can be toggled by selecting "Options" then "Show Toolbar" in the [main menus](#), or by pressing **F4** on the keyboard.

 **Add Torrent** prompts you for the location of the .torrent file that you are trying to open on disk. If you set a [default download location](#), the torrent job will automatically be added to the torrent jobs list and started (if you did not set  $\mu$ Torrent to [start torrent jobs in stopped mode](#)). If you did not set a default download location, you will be prompted where you would like to save the torrent contents, with the [Add New Torrent](#) dialog being shown by default.

 **Add Torrent from URL** allows you to open a .torrent file directly from URL without having saved the it to your hard disk. Depending on the method used by the website for authenticating users (if any), the URL may have to be entered [in a specific format](#). Magnet URIs may be used here, and  $\mu$ Torrent will attempt to download the info dictionary from the corresponding .torrent file from any other client that also supports the URI scheme.

 **Add RSS Feed** opens up the [Add RSS Feed](#) dialog, where you can add RSS feeds for  $\mu$ Torrent to monitor. This is hidden by default in the torrent view, but is shown while viewing Feeds. To show the icon, disable [Narrow Toolbar](#) in the main menu.

---

 **Create New Torrent** lets you [create .torrent files](#) that you can use to share your data with other people.

---

**✖ Remove** will remove the selected torrent job(s) using the default action when **left-clicked**. This default action can be set by **right-clicking** the button, holding **Shift** on the keyboard, and **left-clicking** one of the four possible options. The default remove action can also be performed by pressing **Delete** on the keyboard. By holding **Shift** while pressing **Delete**,  $\mu$ Torrent will also remove the torrent contents in addition to performing the default remove action.

- **Remove** removes the selected torrent job(s) from the list, but all related files are left intact on the disk.
- **Remove and delete .torrent** removes the selected torrent job(s) from the list and the corresponding .torrent file(s) from the **.torrent file storage location**.
- **Remove and delete .torrent + Data** removes the selected torrent job(s) from the list, the corresponding .torrent file(s) from the **.torrent file storage location**, and all content downloaded from the torrent job(s).
- **Remove and delete Data** removes the selected torrent job(s) from the list and all content downloaded from the torrent job(s).
- **Move to trash if possible** tells  $\mu$ Torrent to attempt to move any deleted file to the Recycle Bin first rather than deleting it immediately off the disk. This works only if the Recycle Bin has enough space allocated.

---

**▶ Start** will start the selected torrent job(s), or add it onto the queue if the number of active torrent jobs has reached the maximum set in the Preferences.

**⏸ Pause** will pause the selected torrent job(s), but won't actually stop it. This tells  $\mu$ Torrent to attempt to retain connections to peers without having to re-establish them like starting stopped torrent jobs would require. It is useful for when you need quick access to bandwidth. Realize that while  $\mu$ Torrent won't drop the connections on its own accord, the connection can still get dropped by the client on the other end of the connection.

**■ Stop** will stop the selected torrent job(s). All connections with peers are dropped.

---

**▲ Move Up Queue** will decrease the selected torrent's queue

number, thus bringing it closer to the top of the queue (meaning it will become active sooner than those with higher queue numbers). Holding **Shift** while pressing this button will move the selected torrent job to the top of the queue. Note that this button will not visually move the torrent job up the list unless you sort the list by the "#" column. Pressing **Ctrl + Alt + Up** on the keyboard is equivalent to pressing this button when a torrent job is selected in the torrent jobs list.

♥ **Move Down Queue** will increase the selected torrent's queue number, thus bringing it further from the top of the queue (meaning it will become active later than those with lower queue numbers). Holding **Shift** while pressing this button will move the selected torrent job to the bottom of the queue. Note that this button will not visually move the torrent job down the list unless you sort the list by the "#" column. Pressing **Ctrl + Alt + Down** on the keyboard is equivalent to pressing this button when a torrent job is selected in the torrent jobs list.

---

🌐 The **Search** field allows you to enter a query to be searched on the selected search engine. To perform the search, press the button next to the input field, or press **Enter** on the keyboard. This will open the search results in your default web browser. To change search engines, click on the dropdown arrow next to the search button. If you are using an operating system older than Windows XP, you will need to **right-click** the button to see the search engine selection menu. Note that the search button and search field will be hidden if the **search engines list** is empty.

🔍 The **Torrent Filter** field allows you to search through your list of torrents by specific text. You can access the torrent filter by clicking the small arrow to the right of the search icon and selecting "Filter My Torrents," or by pressing **Control + F**. If you are using an operating system older than Windows XP, you will need to **right-click** the button to see the search engine/torrent filter selection menu. As you enter text your list of torrents will be filtered automatically, and only torrents that match your search text will be

displayed in the main list of torrents. To remove the filter and view all torrents, simply remove any text from the filter box.

 **Toggle Torrent View** will change the main-view look between the more classic list-view and a more graphics-oriented - look & feel.

 **µTorrent Remote** opens up the [Remote](#) dialog, where you can configure most of µTorrent's options.

 **Preferences** opens up the [Preferences](#) dialog, where you can configure most of µTorrent's options.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface >](#)

[Main Window](#)

# Category List

The category list contains various features in the µTorrent client, including the list of torrent jobs, labels, and RSS feeds. The category list's visibility can be toggled by selecting "*Options*" then "*Show Category List*" in the **main menus**, or by pressing **F7** on the keyboard. The last selected category before µTorrent is exited will be automatically selected on the next start. Numbers in the parentheses next to the category name indicate the number of torrent jobs there are listed under the respective category. The treeview is composed of the following default categories:

-  **Featured Content** Replaces the main view with a Web-like pane of featured content.
-  **Torrents** applies no filters and displays all torrent jobs in your client.
-  **Downloading** only shows torrent jobs that are not finished or completed.
-  **Seeding** only shows torrent jobs that were completed and are now seeding.
-  **Completed** displays all torrent jobs that have finished downloading.
-  **Active** only shows torrent jobs that are actively being transferred, whether it is downloading or seeding. Torrent jobs must be downloading at rates above the value specified by `queue.slow_dl_threshold` or uploading at rates above the value specified by `queue.slow_ul_threshold` to be considered active.
-  **Inactive** only shows torrent jobs that are not being actively transferred. This includes all torrent jobs that are stopped, waiting in the queue, or are started, but downloading at rates below the value specified by `queue.slow_dl_threshold` and uploading at rates below

the value specified by `queue.slow_ul_threshold`.

---

-  **Labels** applies no filters and displays all torrent jobs in your client. Underneath this item is the list of all labels on your client.
  -  **No Label** will display all torrent jobs that have no labels set.
- 
-  **Feeds** will display the RSS items from all RSS feeds. Underneath this item is the list of all feeds on your client.
  -  **Devices** will display a view all all the paired-devices that you have added.
- 

User-added labels will appear in the category list immediately below "No Label". Torrent jobs with labels will be displayed under their corresponding label. Standard labels are removed from the category list as soon as there are no torrent jobs using them.

Not all categories are mutually exclusive, so if you have a torrent job that has no labels, and is inactive, it will show up in both the "Inactive" category, and the "No Label" category. Multiple categories can be selected by holding **Shift** or **Ctrl** while **left-clicking** on a category. Holding **Shift** will select all categories in between the first selected category and the last selected one. Holding **Ctrl** allows you to select (or deselect) specific categories that aren't necessarily adjacent to each other.

RSS feeds are displayed at the bottom of the category list (below the labels). When an RSS feed is selected, the torrent jobs list is filled with torrents as listed in the RSS feed. Torrent jobs added from a feed are also listed in their respective source feeds.

## Context Menu

If you **right-click** on an item in the category list, you'll see a context menu. They are described in the following list:

- **Add Torrent...** prompts you for the location of the .torrent file that you are trying to open on disk. If you set a **default download location**, the torrent job will automatically be added to the torrent jobs list and started (if you did not set µTorrent to **start torrent jobs in stopped mode**). If you did not set a default download location, you will be prompted where you would like to save the torrent contents, with the **Add New Torrent** dialog being shown by default.
  - **Add RSS Feed...** opens up the **Add RSS Feed** dialog, where you can add RSS feeds for µTorrent to monitor.
  - **Transfer to..** transfer contents to any of the connected paired-devices.
- 
- **Torrent Options** displays select menu items from the **torrent jobs list context menu**, and behave identically to their torrent jobs list context menu counterparts. These menu items show up only if there are torrent jobs selected in the torrent jobs list.
  - **RSS Feed Options** displays options for the RSS feed, and show up only if RSS items are selected in the torrent jobs list. If "**All Feeds**" item is right-clicked, the selected action is applied on every feed.
    - **RSS Downloader** opens up the **RSS Downloader** dialog.
    - **Update Feed** allows you to perform a manual reload of the selected RSS feed(s). Refrain from using this repeatedly in rapid succession, as hammering the RSS feed's server won't make the feed be updated with new items any more quickly, and you run the chance of causing the server to fail. In short, do not abuse this feature.
    - **Enable/Disable Feed** enables or disables the RSS feed updating for the selected feed(s).
    - **Edit Feed...** opens up the **Edit RSS Feed** dialog, where you can edit the properties of the selected RSS feed(s). If multiple feeds are selected, µTorrent will open an edit dialog for each feed one at a time, opening another dialog for the next feed only after the current feed's edit dialog is closed.
    - **Delete Feed** removes the RSS feed from the list.
    - **Clear Feed History** removes all entries in the RSS Downloader's **History tab** that came from the selected feed(s).
    - **Torrent Options** displays select menu items from the **torrent jobs list context menu**, and behave identically to their torrent jobs list context menu counterparts. These menu

items show up only if there are torrent jobs selected in the torrent jobs list for the relevant RSS feed(s).

[« Previous](#)   [Next »](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Main Window](#)

# Torrent Jobs List

The torrent jobs list is the main interface item in  $\mu$ Torrent. It is where your torrent jobs are displayed, along with plenty of information for each of them. Sorting by more than one column is possible in this list. You can do so by sorting by one column, then hold **Shift** on your keyboard while selecting the secondary column by which  $\mu$ Torrent should sort the list by. If an RSS item is **double-click**ed, it gets added to the download queue. The following is a description of each column:

- **Name** displays the name of the torrent job. The initial name used for each torrent job is the name of the file being transferred (if the torrent contents consist of a single file), or the directory being transferred (if the torrent contents consist of multiple files). A torrent job can be renamed in the list by slow **double-click**ing on the torrent job, or by pressing **F2** on the keyboard while the torrent job is selected. To cancel the renaming, you should press **Esc** on the keyboard. To confirm the change, you should press **Enter** on the keyboard, or use the mouse to click anywhere else on the torrent jobs list. Renaming the torrent job will not rename the file or directory being transferred.
  -  means the torrent job is downloading without issue
  -  means the torrent job is a queued download
  -  means the torrent job is downloading, but there is a tracker error
  -  means the torrent job is seeding without issue
  -  means the torrent job is a queued seed
  -  means the torrent job is seeding, but there is a tracker error
  -  means the torrent job is paused
  -  means the torrent job is stopped, but hasn't finished downloading
  -  means the torrent job is stopped, and has finished downloading
  -  means the torrent job has a critical error (check the Status column)
  -  means the item is an RSS feed item that has not been added to the torrent jobs queue
  -  means the release is less than 24 hours old
  -  means the torrent job has already been added and moved to the history for the RSS item

- **#** displays the torrent job's place in the download queue. When a torrent job stops or finishes, the next lowest numbered torrent job that is queued for download will start automatically. Torrent jobs that have not reached **the seeding goal** will have a \* (asterisk) instead of an integer. Torrent jobs that have reached the seeding goal will have a blank in this column. Forced torrent jobs do not follow the queue order, although they will be assigned queue numbers like regular torrent jobs. Note that if you want the arrow buttons to move the torrent jobs up and down in the list visually, you must sort the torrent jobs list by this column.
- **Added** displays the date and time that the torrent job was added to the torrent jobs list.
- **Availability** displays the number of copies of the torrent contents that are distributed in the swarm.
- **Bandwidth Allocation** displays the bandwidth allocation given to the torrent job (High, Normal, or Low). **Right-click**ing a torrent job on this column will bring up the **Bandwidth Allocation** context sub-menu.
- **Codec** displays the codec that the video was encoded in assuming it is a video and is an RSS item.
- **Completed** displays the total amount of data you have in the torrent contents that passed the hash checks, as well as any incomplete pieces currently in progress.
- **Completed On** displays the date and time that the torrent job finished downloading. If the torrent job has yet to be completed, this column will be blank.
- **Debug** displays information that might be useful in debugging problems.
- **Disk Job** displays information on any torrent jobs that are having disk I/O issues.

- **Done** displays the approximate percent of the torrent job that  $\mu$ Torrent has completed. Its value is calculated by dividing the number under the **Completed** column by the number under the **Selected Size** column. If `gui.graphic_progress` is enabled,  $\mu$ Torrent will draw a progress bar behind the percentage. If `gui.piecebar_progress` is enabled,  $\mu$ Torrent will draw a piece progress bar instead.
- **Down Limit** displays the maximum download rate limit set on the individual torrent job. If no limit is set, this column will be blank. Right-clicking a torrent job on this column will bring up the **Set Download Limit** context sub-menu.
- **Down Speed** displays the current speed at which data is being downloaded for the individual torrent job. If the download rate is negligible, this column will be blank.
- **Downloaded** displays the total amount of data you have downloaded, including those that were wasted or failed the hash checks.
- **Elapsed** displays the total amount of time that  $\mu$ Torrent has had the torrent job started for since it was added.
- **Episode** displays the episode number of the torrent job, if applicable.
- **ETA** displays the estimated amount of time needed for  $\mu$ Torrent to finish downloading the torrent job. If the torrent job is in seeding mode, then this column displays the estimated time it will take for  $\mu$ Torrent to reach **the seeding goal**. If the torrent job is stopped or has reached its seeding goal, this column will be blank. If the torrent job is transferring too slowly, this column will display the  $\infty$  symbol. Note that this column takes selective file downloading into account, so only the data you select to be downloaded will be counted in the time calculation.
- **Format** displays the video quality of the torrent job, assuming it is a

video and is an RSS item.

- **Health** displays 0-5 bars that indicate the availability of the torrent using: this formula -  $\log_2(\text{Availability})$ . I.e. distributed copies being 1 = 1 bar, 2 = 2 bars, 4 = 3 bars, 8 = 4 bars and 16 = 5 bars.
- **Label** displays the label set for the torrent job. Right-clicking a torrent job on this column will bring up the **Label** context sub-menu.
- **Last Active** displays the amount of time that has elapsed since  $\mu$ Torrent last finished transferring a chunk of data for the torrent job. When the torrent job is downloading, this time will be reset only upon the download completion of a piece. When the torrent job is seeding, this time will be reset only upon the upload completion of a 16 KiB block of data. This number is updated only when the torrent job is started. It should be understood that for stopped torrent jobs, the number is relative to the time at which the torrent job was actually stopped.
- **Peers** displays the number of peers you are connected to, and the number of peers in the swarm within the parentheses. The number of peers in the swarm is an estimate based on the maximum number of peers found either between the amounts reported by any tracker, or the number of peers  $\mu$ Torrent has encountered within its peer cache. If the torrent job is stopped, and **bt.scrape\_stopped** is enabled,  $\mu$ Torrent will show the maximum number of peers in the swarm as reported by the tracker scrapes.
- **Playback** - the multi-purpose stream/play/status button and display lets you start streaming (of your choice of media file), play the stream when ready and in between - shows the estimated time till this stream is ready to play.
- **Ratio** displays the ratio of uploaded data to downloaded data. Its value is calculated by dividing the number under the **Uploaded** column by the number under the **Downloaded** column. If the torrent job was added when the file was already complete (seeding the file upon adding), then the ratio will start from 0, and will be calculated

based on the ratio of the uploaded data to the torrent content size.

- **Rating** displays the average scores given by all users to this torrent
- **Remaining** displays the amount of data left for  $\mu$ Torrent to download before it finishes downloading the torrent job. Note that this column takes selective file downloading into account, so only the data you select to be downloaded will be counted. If no data is left to be downloaded, this column will be blank.
- **Seeds** displays the number of seeds you are connected to, and the number of seeds in the swarm within the parentheses. The number of seeds in the swarm is an estimate based on the maximum number of seeds found either between the amounts reported by any tracker, or the number of seeds  $\mu$ Torrent has encountered within its peer cache. If the torrent job is stopped, and `bt.scrape_stopped` is enabled,  $\mu$ Torrent will show the maximum number of seeds in the swarm as reported by the tracker scrapes.
- **Seeds/Peers** displays the ratio of seeds to peers. Its value is by dividing the number in the parentheses under the `Seeds` column by the number in the parentheses under the `Peers` column.
- **Selected Size** shows only the size of the torrent contents you have selected for  $\mu$ Torrent to download. The size of all pieces belonging to skipped files that also pass the checks will also be taken into account and added in this column.
- **Size** shows the complete size of the torrent contents, regardless of whether you are using selective file downloading on that torrent job or not.
- **Source URL** displays the URL that the source .torrent file is located at. The source URL is only applicable for RSS items and torrent jobs `added from URL`.
- **Status** displays the current status of the torrent job. If an error occurs on this torrent job, depending on the error, it might be

displayed on this column. Otherwise, the following are normally displayed:

- **Downloading** means the torrent job is currently downloading.
  - **Finished** means the torrent job has been stopped after it reached seeding mode. If the torrent job reaches this without user intervention, then it means it reached [the seeding goal](#).
  - **Initial-Seeding** means the torrent job is being seeded in Initial Seeding mode.
  - **Paused** means the torrent job is paused, but not stopped (still connected to peers).
  - **Queued** means the torrent job is waiting for another torrent job to finish downloading before starting.
  - **Queued Seed** means the torrent job is waiting for another torrent job to finish seeding before starting.
  - **Seeding** means the torrent job is finished downloading, and is currently in seeding mode.
  - **Stopped** means the torrent job is stopped (not transferring, and not connected to any peers).
  - **[F] Downloading** means the torrent job is forced started and downloading, so it is outside of the queue order.
  - **[F] Initial-Seeding** means the torrent job is forced started and seeding in Initial Seeding mode, so it is outside of the queue order. Forced initial-seeding torrent jobs do not obey the seeding goal.
  - **[F] Seeding** means the torrent job is forced started and seeding, so it is outside of the queue order. Forced seeding torrent jobs do not obey the seeding goal.
  - **Previously Downloaded** means the item is a part of an RSS feed, and was previously added to the torrent jobs list (it is listed in the [RSS History](#) tab).
  - **Waiting for Metadata** applies to magnet links and means that µTorrent is waiting to download the meta details about the files contained within the magnet link.
  - **Connecting to Peers** means the client is waiting to make a connection to at least one peer so it can begin downloading the file.
  - **Checked %** means the client is currently checking previously-downloaded data for the torrent job in question.
  - **RSS** means the item is a part of an RSS feed, and not yet added to the torrent queue.
  - **Moving** means the torrent job has completed, and the torrent file(s) are being moved to the alternate destination path. This status will also show up when using the "Set Download Location" function.
  - **Flushing** means the torrent is about to finish downloading and is writing the pieces, that are still in cache - to the disk.
- **Tracker** displays the first working tracker's base URL.

- **Tracker Status** displays the tracker's status.
- **Up Limit** displays the maximum upload rate limit set on the individual torrent job. If no limit is set, this column will be blank. Right-clicking a torrent job on this column will bring up the **Set Upload Limit** context sub-menu.
- **Up Speed** displays the current speed at which data is being uploaded for the individual torrent job. If the upload rate is negligible, this column will be blank.
- **Uploaded** displays the total amount of data you have uploaded.

## Context Menu

If you right-click on a torrent job, you'll see a context menu. They are described in the following list:

- **Torrent Options**
  - **Transfer** will copy your converted media file(s) to your selected paired device
  - **Open** will attempt to open the torrent contents for the selected torrent job(s) with the associated application for the file type. This will only work for single-file torrent jobs.
  - **Open Containing Folder** opens the directory containing the torrent contents in an Explorer window. If the torrent job contents consist of a single file, that file will be highlighted. Otherwise, the torrent job's directory will be displayed instead.
  - **Show Converted Files** will open Windows Explorer in the location of the media files converted for your paired device
  - **Copy Magnet URI...** copies a magnet URI link to the Windows clipboard for the selected torrent job(s), multiple URIs separated by newlines.
  - **Open URL in Browser** will open the linked URL(s) in your default web browser.
  - **Force Start** takes the selected torrent job(s) out of queue order (makes it ignore **queueing**) and forces it to start immediately, even if the maximum number of active torrent jobs and/or downloads has been reached. Forced torrent jobs will not stop even if they reach the seeding goal. The **scheduler** is unable to stop forced torrent jobs.
  - **Start** will start the selected torrent job(s), or add it onto the queue if the number of active torrent jobs has reached the maximum set in the Preferences.
  - **Pause** will pause the selected torrent job(s), but won't actually stop it. This tells µTorrent to attempt to retain connections to peers without having to re-establish them like starting stopped torrent jobs would require. It is useful for when you need quick

access to bandwidth. Realize that while  $\mu$ Torrent won't drop the connections on its own accord, the connection can still get dropped by the client on the other end of the connection.

- **Stop** will stop the selected torrent job(s). All connections with peers are dropped.
- **Move Up Queue** will decrease the selected torrent's queue number, thus bringing it closer to the top of the queue (meaning it will become active sooner than those with higher queue numbers). Holding **Shift** while pressing this button will move the selected torrent job to the top of the queue. Note that this button will not visually move the torrent job up the list unless you sort the list by the "#" column. Pressing **Ctrl** + **Alt** + **Up** on the keyboard is equivalent to selecting this item when the torrent job is selected.
- **Move Down Queue** will increase the selected torrent's queue number, thus bringing it further from the top of the queue (meaning it will become active later than those with lower queue numbers). Holding **Shift** while pressing this button will move the selected torrent job to the bottom of the queue. Note that this button will not visually move the torrent job down the list unless you sort the list by the "#" column. Pressing **Ctrl** + **Alt** + **Down** on the keyboard is equivalent to selecting this item when the torrent job is selected.
- **Labels** allows you to set or reset labels for any selected torrent job. All labels will be listed at the this submenu, where you can also select-check (or un-check) any of them for this torrent.
  - **New Label...** will display a dialog asking you to enter the name of a new (temporary) label.
  - **Set Primary** will choose a label to be the first one displayed (primary) for this torrent.
- **Bandwidth Allocation** is an option that makes  $\mu$ Torrent allocate upload bandwidth to the selected torrent job(s) based on the option selected. This option works only if a **global maximum upload rate** is set, or the selected torrent job(s) each have an individual **maximum upload rate** set.
  - **High** will give the selected torrent job(s) more upload bandwidth relative to other torrent jobs of lower bandwidth allocation levels (Low or Normal).
  - **Normal** is the default bandwidth allocation given.
  - **Low** will give the selected torrent job(s) less upload bandwidth relative to other torrent jobs of higher bandwidth allocation levels (Normal or High).
  - **Set Download Limit** allows you to control the maximum download rate for the selected torrent job(s). The values displayed depend on **the speed popup list settings** in the Preferences. If the speed popup list is not manually overridden, then the maximum value available for selection will depend on the **global maximum download rate limit**.
  - **Set Upload Limit** allows you to control the maximum upload rate for the selected torrent job(s). The values displayed depend on **the speed popup list settings** in the Preferences. If the speed popup list is not manually overridden, then the maximum value available for selection will depend on the **global maximum upload**

rate limit.

---

- **Remove** removes the selected torrent job(s) from the list, but all related files are left intact on the disk.
  - **Remove And**
    - **Delete .torrent** removes the selected torrent job(s) from the list and the corresponding .torrent file(s) from the **.torrent file storage location**.
    - **Delete .torrent + Data** removes the selected torrent job(s) from the list, the corresponding .torrent file(s) from the **.torrent file storage location**, and all content downloaded from the torrent job(s). This action can also be performed by pressing **Shift + Delete** on the keyboard.
    - **Delete Data** removes the selected torrent job(s) from the list and all content downloaded from the torrent job(s).
  - **Force Re-Check** tells  $\mu$ Torrent to check the torrent contents of the selected torrent job(s) for missing files and pieces that fail the hash check. Even if multiple torrent jobs are selected to be re-checked,  $\mu$ Torrent will check them sequentially, one at a time (known as a round-robin method). Pausing a torrent job while it is being re-checked will pause the re-checking process for that torrent job, and will move onto the next torrent job to re-check if multiple torrent jobs are selected. Any incomplete pieces in the torrent contents will be removed during the re-checking process, so a drop in progress percentage would not be an unlikely occurrence.
  - **Advanced**
    - **Reset Bans** allows you to clear any bans  $\mu$ Torrent has placed on peers in the selected torrent job(s) for sending too many hashfailed pieces.
    - **Clear Peer List** allows you to clear all peers from the peer list for the selected torrent job(s).
    - **Set Download Location...** lets you set the location of the contents for the particular torrent if they are ever moved after the torrent job is added. The next time the torrent job is started, it will be hash checked to verify the data integrity.
    - **Set Destination Name...** lets you change the name of the directory into which the files will be downloaded.
    - **Show Download Bar** toggles the display of a floating download bar that shows quick statistics for the selected torrent job(s). The download bars are separate from the  $\mu$ Torrent interface, so they can be displayed while the main window is minimized or closed to tray.
  - **Update Tracker** allows you to perform a manual announce on the selected torrent jobs' trackers. Only trackers whose minimum intervals have elapsed since the last manual update will be updated. This option is disabled if 60 seconds have not yet passed since the last time it was used, or if the elapsed time since the last update is less than the minimum interval for every selected torrent job's tracker(s). If the condition to be enabled has been satisfied by at least one of the selected torrent jobs, the option will be enabled, but when used, will be invoked only on those torrent jobs that have met the conditions for option to be enabled.
-

- **Properties** opens up the [Torrent Properties](#) dialog, where you can modify several of the torrent job's settings.

- **RSS Item Options**

- **Open** will attempt to open the .torrent file(s) associated with the selected item(s) to be added to the torrent jobs list. [Double-click](#)ing on the selected item achieves the same result.
- **Open URL in Browser** will open the linked URL(s) in your default web browser. This is useful for RSS feeds that do not link directly to .torrent files, but instead, to a page that links to them.
- **Clear Feed History** removes all entries in the RSS Downloader's [History tab](#) that came from the selected feed(s).
- **Add to Favorites** adds the selected items to the [Favorites](#), attempting to create suitable rules for the selected items.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface >](#)

[Main Window](#)

## Detailed Info Pane

The detailed info pane allows you to view detailed information about the selected torrent job that isn't available in the torrent job list itself. There are several tabs, each of which contain different types of information. The detailed info pane's visibility can be toggled by selecting "Options" then "Show Detailed Info" in the **main menus**, or by pressing **F5** on the keyboard. The last selected tab before  $\mu$ Torrent is exited will be automatically selected on the next start.

-  Files
-  Info
-  Peers
-  Ratings
-  Trackers
-  Pieces
-  Speed
-  Logger
-  Related

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Files

■ The Files tab allows you to view what files are included in the torrent, as well as provide the ability to control what files are downloaded. **Double-click**ing on any file will open it as if it were opened from Explorer (meaning that  $\mu$ Torrent will attempt to open the file in the associated application for the file type).

- **Name** is the file's original path as described in the .torrent file metadata.
- **Path** is the file's save path on disk. If the path is not an absolute (full) path, it is the path relative to the torrent job's **Save As** directory.
- **# Pieces** is the number of pieces that comprise the file.
- **%** is the approximate percent of the file that  $\mu$ Torrent has completed. Its value is calculated by dividing the number under the **Done** column by the number under the **Size** column. Because this number is calculated using the size of all completed pieces only, it will not increase until another piece of the file has finished downloading.
- **Done** displays the total amount of data you have downloaded for that file that passed the hash checks.
- **First Piece** is the number of the first piece in the .torrent file that contains data for the file.
- **Mode** displays how  $\mu$ Torrent is using the file.
  - **Write** mode means the file is being actively written to.
  - **Read** mode means the file is not being actively written to.
- **Pieces** is the graphical progress bar for the file.
  - **Blue** means the data has been written to disk.
  - **Green** means the data is still unwritten to disk.
  - **Red** means the data is is not available in the swarm.
  - **White** means the data is is available in the swarm, but has not yet been downloaded.

- **Priority** displays the priority given to the file.
  - **high** means  $\mu$ Torrent will attempt to complete the file more aggressively relative to other files of lower priorities (`low` or `normal`). If there is an opportunity for this file to become more complete,  $\mu$ Torrent will take it. Note that this does not guarantee the completion of the file, or that the file will complete before other files of lower priorities.
  - **normal** is the default priority given.
  - **low** means  $\mu$ Torrent will attempt to complete the file less aggressively relative to other files of higher priorities (`normal` or `high`). If there is an opportunity for another file of higher priority to become more complete,  $\mu$ Torrent will take it over downloading for this file. Note that this does not guarantee that the file will complete after other files of higher priorities.
  - **skip** means the file will not be downloaded. Do note that some files share pieces with other files. As such, you might notice that a part of a skipped file (or possibly an entire file, if it is smaller than the piece size) is downloaded. Understand that this does not mean  $\mu$ Torrent has ignored your request to not download the file.
- **Size** shows the size of the file.
- **Rate, Resolution, Duration, Streamable, Has Header, Codecs** - all of those are internal properties related to media files, that might be streamed.

## Context Menu

- **Open** will attempt to open the selected file as if it were opened from Explorer.
  - **Open Containing Folder** opens the file's directory containing in an Explorer window, with the file highlighted.
- 
- **High Priority** will set the file's priority to `high`.
  - **Normal Priority** will set the file's priority to `normal`.
  - **Low Priority** will set the file's priority to `low`.
- 
- **Don't Download** will set the file's priority to `skip`.

- 
- **Relocate...** will set the file to be downloaded to a location other than the location selected for the entire torrent job. This option is disabled if the torrent job is started.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Info

**i** The General tab is where you'll find most of the general statistics for the currently selected torrent job available.

## Graphical Progress Bars

- The **upper Downloaded bar** is the general progress bar useful for getting a quick visual outlook of the percent completed. Selective file downloading is taken into account, so only the data you select to be downloaded will be counted. The number to the right of the Downloaded bar is the percentage of the torrent job that is completed, taking selective file downloading into account as well.
  - **White** represents what has not yet been downloaded.
  - **Blue** represents what has been downloaded.
- The **lower Downloaded bar** is equivalent to the upper bar, except that it does not take selective file downloading into account, and it also displays the distribution of pieces completed in your torrent job. If there are more pieces in the .torrent file than there are horizontal pixels for your desktop resolution, µTorrent will take the average completion of adjacent pieces to decide what colors should be shown.
  - **White** represents the pieces that have not yet been downloaded.
  - **Blue** represents the pieces that have been downloaded.
- The **Availability bar** shows the spread of the pieces in the network of peers you are connected to. If there are more pieces in the .torrent file than there are horizontal pixels for your desktop resolution, µTorrent will take the average completion of adjacent pieces to decide what colors should be shown. The number to the right is the actual availability of the selected torrent job.
  - **Dark blue** means the piece is available and common.
  - **Light blue** means the piece is available, but rare.  
torrent content size
  - **Red** means the piece is not available within the group of peers you are connected to.

Along with graphical progress bars, textual statistics are provided below. Any of these fields can be copied by **right-click**ing it and selecting "*Copy*". The following is a description of the information displayed (note that most of the information displayed here can be found in the torrent jobs list):

## Transfer

- **Time Elapsed** displays the total amount of time accumulated that the torrent job has been started.
- **Remaining** displays the estimated amount of time needed for  $\mu$ Torrent to finish downloading the torrent job. If the torrent job is in seeding mode, then the ETA displays the estimated time it will take for  $\mu$ Torrent to reach the seeding goal. If the torrent job is stopped, this field will be blank. If the torrent job is transferring too slowly, or has reached its seeding goal, this field will display the  $\infty$  symbol. Note that this field takes selective file downloading into account, so only the data you select to be downloaded will be counted.
- **Wasted** displays the amount of data that was downloaded and deleted, either from hashfails, or from redundant data. The number in the parentheses is the count for the number of hashfails that occurred.
- **Downloaded** displays the total amount of data you have downloaded, including those that were wasted or failed the hash check.
- **Uploaded** displays the total amount of data you have uploaded since the torrent job was added.
- **Seeds** displays the number of seeds you are connected to. The second number is the number of known seeds in the peer cache collected from the tracker, DHT, and PEX. The number in the parentheses is the number of seeds reported by the tracker scrape.
- **Download Speed** displays the current speed at which data is being

downloaded for the individual torrent job. The number in the parentheses is the average download speed throughout the time that the torrent job was in Downloading mode.

- **Upload Speed** displays the current speed at which data is being uploaded for the individual torrent job. The number in the parentheses is the average upload speed throughout the time that the torrent job was started.
- **Peers** displays the number of peers you are connected to. The second number is the number of known peers in the peer cache collected from the tracker, DHT, and PEX. The number in the parentheses is the number of peers reported by the tracker scrape.
- **Down Limit** displays the maximum download rate limit set on the individual torrent job. If no limit is set, or the limit is set to 0, this field will display the  $\infty$  symbol.
- **Up Limit** displays the maximum upload rate limit set on the individual torrent job. If no limit is set, or the limit is set to 0, this field will display the  $\infty$  symbol.
- **Share Ratio** displays the ratio of uploaded data to downloaded data. You can calculate it by dividing the number beside the **Uploaded** field by the number beside the **Download** field. If the torrent job was added when the file was already complete (seeding the file upon adding), then the ratio will start from 0, and will be calculated based on the ratio of the uploaded data to the torrent content size.
- **Status** displays the status of that torrent as seen on the status column.

## General

- **Save As** shows the location that the torrent contents are saved to on the disk.

- **Total Size** shows the total size of the torrent contents. The first number displays the total amount of data in the torrent contents, taking selective file downloading into account. The second number is the total amount of data you have downloaded that passed the hash checks and have been written to disk. If there is a third number here, then it's because you are using selective file downloading. That third number displays the total size of the torrent contents, not taking selective file downloading into account. **Comment** shows the comment placed in the .torrent file by the .torrent file's creator.
- **Pieces** shows the number of pieces in the torrent job, and the size of each of these pieces. The number in the parentheses indicates the number of pieces  $\mu$ Torrent has successfully acquired.
- **Created On** shows the date and time that the .torrent file was created along with the application that was used to create the .torrent file (if it can be determined).
- **Created By** shows whom the .torrent file was created by (if it was entered into it).
- **Added On** shows the date and time that the .torrent file was added by the user onto the torrents' jobs list.
- **Completed On** shows the date and time that the torrent job has been completed.
- **Hash** shows the .torrent file's info dictionary hash.
- **Comment** shows any comment that was entered into the .torrent file.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Ratings

👉 In the Ratings tab you can view all the comments other users have given to this torrent. You can also add your own comment.

- **Submit** - in this edit box you can edit and submit your own remarks about this torrent.

[« Previous](#)   [Next »](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Trackers

✖ The Trackers tab displays the trackers that  $\mu$ Torrent is currently using for the selected torrent job. The following is a description of each column:

- **Name** displays the tracker URL, or the name of the alternative sources for obtaining peers.
- **Downloaded** displays the number of times the torrent job was reported to be complete by peers on the tracker.
- **Interval** displays the amount of time the tracker recommended  $\mu$ Torrent to wait before next automatic announce.
- **Min Interval** displays the minimal amount of time the tracker requires  $\mu$ Torrent to wait before next announce. This value is dictated by the tracker, and may be different from tracker to tracker. If the tracker specifies a minimum announce interval of less than 30 seconds,  $\mu$ Torrent will use 30 seconds as the minimum interval instead.
- **Peers** displays the number of peers returned by the tracker scrape.
- **Seeds** displays the number of seeds returned by the tracker scrape.
- **Status** shows the current status of the tracker. If an error occurs while attempting to use the tracker, depending on the error, it might be displayed on this column. The following are commonly displayed:
  - **disabled** means the source will not be used for the torrent job due to the options selected in [the torrent properties](#). This occurs only with DHT, Local Peer Discovery, and Peer Exchange.
  - **hostname not found** means the tracker IP could not be resolved. This may occur due to DNS issues, or because the domain or hostname specified is invalid or does not exist.
  - **invalid url** means the tracker specified is not valid, and cannot be used with  $\mu$ Torrent. This most often occurs if the specified tracker is not of the correct protocol ( $\mu$ Torrent

supports only HTTP, HTTPS, and UDP trackers).

- **not allowed** means the source cannot be used with the torrent job. This status occurs only with DHT, Local Peer Discovery, and Peer Exchange, and only when the torrent job is private.
  - **scrape ok** means the tracker scrape successfully returned the seed and peer counts for the torrent job.
  - **scrape not supported** means the tracker does not support scraping.
  - **working** means the tracker responded properly on the last tracker update.
- **Update In** displays the amount of time (in minutes) until  $\mu$ Torrent next announces to the tracker.

## Context Menu

- **Update Tracker** allows you to perform a manual announce on the selected trackers, assuming the minimum announce interval has passed for the trackers. This option is disabled if 60 seconds have not yet passed since the last time it was used, or if the elapsed time since the last update is less than the minimum interval for every selected tracker. If the condition to be enabled has been satisfied by at least one of the selected trackers, the option will be enabled, but when used, will be invoked only on those trackers that have met the conditions for option to be enabled.
  - **Remove Tracker** removes the selected tracker(s) from the tracker list. This action can also be performed by pressing **Delete** on the keyboard.
- 
- **Add Tracker...** opens the **torrent properties** dialog, where you can add trackers to the tracker list.
  - **Use DHT** is a shortcut for the **Enable DHT** option in the torrent properties.
  - **Use Local Peer Discovery** is a shortcut for the **Local Peer Discovery** option in the torrent properties.
  - **Use Peer Exchange** is a shortcut for the **Peer Exchange** option in

the torrent properties.

[« Previous](#)   [Next »](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Peers

 The Peers tab is where you'll find information about all the peers you are currently connected to on the selected torrent. The following is a description of each column:

- **IP** displays the peer's IP address and the flag of the country they're from. If **Resolve IPs** is enabled, the country flag and hostname will be displayed instead.
- **%** displays the approximate percent of the torrent job the peer has completed.
- **Client** displays the BitTorrent client that the peer reports it is using. If µTorrent suspects the client of spoofing (faking) its client ID, it will note it as such.
- **Debug** displays information that might be useful in debugging problems.
- **Down Speed** is the averaged speed at which you are downloading data from the peer.
- **Downloaded** is the amount of data you have downloaded from the peer since the torrent job was added.
- **Flags** displays various letters, each carrying a special meaning about the state of the connection:
  - **?**: your client unchoked the peer but the peer is not interested
  - **D**: currently downloading from the peer (interested and not choked)
  - **d**: your client wants to download, but peer doesn't want to send (interested and choked)
  - **E**: peer is using Protocol Encryption (all traffic)
  - **e**: peer is using Protocol Encryption (handshake)
  - **F**: peer was involved in a hashfailed piece (not necessarily a bad peer, just involved)
  - **H**: peer was obtained through DHT

- **h**: peer connection established via UDP hole-punching
  - **I**: peer established an incoming connection
  - **K**: peer unchoked your client, but your client is not interested
  - **L**: peer has been or discovered via [Local Peer Discovery](#)
  - **O**: optimistic unchoke
  - **P**: peer is communicating and transporting data over uTP
  - **S**: peer is snubbed
  - **U**: currently uploading to the peer (interested and not choked)
  - **u**: the peer wants your client to upload, but your client doesn't want to (interested and choked)
  - **X**: peer was included in peer lists obtained through Peer Exchange (PEX)
- **Hasherr** is the number of pieces this peer sent to you that failed the hash check.
  - **Inactive** displays the number of seconds since the last non-keepalive message was sent with this peer.
  - **MaxDown** is the peak, unaveraged speed at which you were downloading from the peer.
  - **MaxUp** is the peak, unaveraged speed at which you were uploading to the peer.
  - **Peer Download Rate** is an estimated rate at which the peer is downloading based on the peer's reported change in pieces obtained. This estimation is very crude and is most likely inaccurate, so it should only be lightly relied upon.
  - **Port** displays the listening port that the peer is using.
  - **Queued** displays the number of bytes requested by a peer that was left unsent by your client.
  - **Relevance** is the percent of the torrent contents that the peer has that you currently do not have.
  - **Reqs** displays the number of 16 KiB block requests you've made to

the peer on the left side, and the number of 16 KiB requests they've made to you on the right side.

- **Up Speed** is the averaged speed at which you are uploading data to the peer.
- **Uploaded** is the amount of data you have uploaded to the peer since the torrent job was added.
- **Waited** displays the number of seconds since the last request was made to this peer.

## Context Menu

- **Add Peer...** allows you to add a peer manually if you know the peer's IP address and port.  $\mu$ Torrent will attempt to connect to the newly-added peer as soon as possible. Only one peer can be added at a time, in the following format: `IP : port`
    - **IPv4** IPs must be inputted in standard dot-decimal notation, like so: `127.0.0.1:80`.
    - **IPv6** IPs must be inputted in colon-hexadecimal notation (enclosed between square brackets), like so: `[0000:0000:0000:0000:0000:0000:0000:0001]:80`. Zero-grouped IPv6 addresses are accepted.

---
  - **Copy Peer List** will copy the IP addresses and the ports of every peer it has in its peer cache to the clipboard, one `IP : port` pair per line.
  - **Copy Selected Hosts** will copy the IP addresses and the ports of each selected peer, one `IP : port` pair per line.
  - **Log Traffic to Logger Tab** tells  $\mu$ Torrent to log connection details from the torrent job to the [Logger tab](#).
  - **Reload IPFilter** forces  $\mu$ Torrent to reload `ipfilter.dat`. This option is disabled when `ipfilter.enable` is disabled.
- 

- **Resolve IPs** tells  $\mu$ Torrent to convert the IP addresses into

hostnames for display purposes. This option must be enabled in order for flags to be displayed next to the peer.

[« Previous](#)   [Next »](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Pieces



The Pieces tab displays each piece currently being downloaded by  $\mu$ Torrent. If a piece is selected, then holding **Shift** on the keyboard while **right-click**ing on the pieces list will display a context menu with the option, "*Log piece info...*" When selected, debug information is added to the **Logger tab** for the currently-highlighted piece. The following is a description of each column:

- **#** is the piece's number in the .torrent file.
- **# of Blocks** is the number of 16 KiB blocks that compose that single piece.
- **Availability** is the number of times this piece is seen completed within the group of peers you are currently connected to.
- **Blocks** is the graphical progress bar for the piece. You'll notice that there are different colors displayed for different blocks. They are as follows:
  - **Dark blue** means the data has been downloaded and written to disk.
  - **Medium blue** means the data has been downloaded, but remains in the cache in memory, unwritten to disk.
  - **Light blue** means the data has been requested from a single peer.
  - **Green** means the data has been requested from multiple peers. Blocks only turn green during endgame mode.
  - **White** means the data is available in the swarm, but has not yet been downloaded.
- **Completed** is the number of blocks completed in the piece.
- **Mode** is the general evaluation of the speed at which the piece is being transferred by the peer. A piece's mode can be either fast, medium, or slow.  $\mu$ Torrent uses this evaluation to determine whether a piece should be taken from fast peers or slow seeds. This is meant to try to prevent situations where a slow seed sends the same piece as a fast peer. Instead of allowing that to happen,

µTorrent attempts to have the slower seed send another piece, possibly something that might be rarer in the swarm.

- **Priority** shows the priority µTorrent has assigned the piece based on **the priority of the file(s) it belongs to**. If the piece belongs to multiple files, then it is assigned the highest priority of the files it belongs to. If a piece passes the criteria for being given higher priority by **bt.prio\_first\_last\_piece** (assuming the option is enabled), and the piece's priority is not skipped, then piece's priority is elevated by one priority level.
- **Size** is the size of the piece.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Speed

 The Speed tab displays graphs of the the global upload and download speeds, as well as some basic statistics about disk I/O and cache usage.

- The **Show** dropdown menu allows you to select the type of graph you'd like to be displayed in the graph canvas below.
  - **Upload & Download** shows a combination of the **Download** and **Upload** graphs.

---
  - **Download** shows a graph of the current global download speed.
    - The **light green**-colored line represents the global download rate limit. If no limit is set for the global download rate, this line will not be shown.
    - The **solid green**-colored line represents the current global download rate, counting only payload data (as opposed to including overhead).
    - The **long-dashed green**-colored line represents the global download rate for payload data downloaded from local peers.
    - The **short-dashed green**-colored line represents the global download rate, including download overhead. If `gui.graph_overhead` is disabled, this line will not be shown.

---
  - **Upload** shows a graph of the current global upload speed.
    - The **light red**-colored line represents the global upload rate limit. If no limit is set for the global upload rate, this line will not be shown.
    - The **solid red**-colored line represents the current global upload rate, counting only payload data (as opposed to including overhead).
    - The **long-dashed red**-colored line represents the global upload rate for payload data uploaded to local peers.
    - The **short-dashed red**-colored line represents the global upload rate, including upload overhead. If `gui.graph_overhead` is disabled, this line will not be shown.
    - The **solid yellow**-colored line represents the current upload speed going toward non-BitTorrent receivers, like media players streaming media from µTorrent, or browsers requesting data from µTorrent's **Web UI** backend.

---
  - **Disk Statistics** shows several graphs pertaining to disk and cache access. In each of these graphs, the lighter lines represent the graphs for writes or reads (depending on which graph you're looking at) to and from the cache. Their more solid counterparts represent the writes and reads to and from the disk. Textual statistics pertaining to disk and cache access collected for the current session are shown to the left of the graphs. The "Reset" button below resets the collected statistics.

---
  - **Transfer Cap** shows the cumulative amount of data transferred over the period of time specified in the **Transfer Cap** settings.

- The **solid green**-colored line represents the cumulative amount of data downloaded.
  - The **solid red**-colored line represents the cumulative amount of data uploaded.
  - The **solid yellow**-colored line represents the total cumulative amount of data transferred (downloaded and uploaded).
  - The **light red**-colored line represents the transfer cap limit.
- 
- **Transfer History** shows the amount of data transferred for each day in the last 31 days.
    - The **solid green**-colored line represents the amount of data downloaded.
    - The **solid red**-colored line represents the amount of data uploaded.
    - The **solid yellow**-colored line represents the total amount of data transferred (downloaded and uploaded).
  - **Streaming** shows several graphs related to streaming mechanism for media files in torrents you are now streaming
  - **Network Overhead** shows you several graphs related to all the none-data related traffic introduced by the BitTorrent transfer protocol.
  - **uTP Delay** shows you several graphs for the different timing delay only relevant to the uTP (UDP) type connections.
  - **Disk Timing** shows information about various types of disk activity for a specific torrent.
- 
- The **Resolution** dropdown menu allows you to select the interval for the time unit used for the X-axis of each graph. The Y-axis values are average over this time resolution.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Logger

 The Logger tab displays logged information regarding certain events in  $\mu$ Torrent. Information displayed mostly serve debug purposes, and can be useful when trying to solve problems. Information displayed can be selected, and selected lines can be copied by pressing **Ctrl + C** on the keyboard. You can access some options for controlling the Logger tab by **right-click**ing anywhere on the white area in the tab. You'll be presented with the following items in the context menu:

- **Peer Traffic Logging** contains a set of options that control what messages are displayed in the Logger tab pertaining to peer traffic and communication.
  - **Error Logging** contains a set of options that control what messages are displayed in the Logger tab pertaining to errors that occur during operation.
  - **Verbose Logging** contains a set of options that control what additional, non-critical messages are displayed in the Logger tab.
  - **Clear all logging flags** disables all of the logging options.
- 
- **Copy** copies the selected Logger lines to the clipboard.
- 
- **Log to file...** allows you to save log information to a file on disk. A full path should be specified, otherwise the log's location will be saved to the current working directory, which can vary depending on how you opened  $\mu$ Torrent. In order to log to disk successfully, you must first select this menu item and specify the filename for the log.  $\mu$ Torrent will start the log on disk from that time forward. When you wish to stop logging, select this menu item again and leave the field blank, and press "OK" (unless you wish to save to start writing to a new file in serial). No logger information will be written to the log file

until the logging to disk is stopped.

---

- **Dump DHT Buckets, Dump DHT Tracked, Dump Memory Info, dump Network Info** and **Dump Sockets** dump advanced debug information into the Logger.
- 
- **Clear Log** does just that -- it clears the contents of the Logger tab.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface >](#)

[Main Window](#)

# Status Bar

The status bar shows general information and statistics about  $\mu$ Torrent. Its visibility can be toggled by selecting "Options" then "Show Status Bar" in the [main menu](#), or by pressing **F6** on the keyboard. The leftmost section of the status bar is an empty slot that resizes along with the window's width, so the following descriptions start from the second item from the left and moves rightward.

- The **Notification** section displays messages in  $\mu$ Torrent about the status of the torrent jobs. If there are no issues, this section will remain blank.
  - **Disk Overloaded** means that the disk was not able to keep up with the read/write speed. If this happens when you first add a torrent job, it's normal, and will disappear after several minutes. If otherwise, you may want to try tweaking your [Disk Cache](#) settings in the Preferences
  - **Limited by scheduler** means the download and/or upload rates are following the [Scheduler](#) rules for the hour.
  - **Stopped by scheduler** means all unforced torrent jobs are stopped to observe the [Scheduler](#) rules for the hour.
  - **Stopped by speed test** means all started torrent jobs, forced or otherwise, are stopped to allow the [Setup Guide's](#) speed tester to obtain as accurate a result as it can achieve. All torrent jobs stopped by the speed test are resumed after the speed test completes.
  - **Stopped by transfer cap** means all unforced torrent jobs are stopped to observe the [Transfer Cap](#) rules.
  - **Seeding only** means any unforced torrent job that is not seeding are stopped to observe the [Scheduler](#) rules for the hour.

---

- The **DHT Status** section displays the current status of DHT.
  - **DHT: Disabled** means you have disabled DHT, so  $\mu$ Torrent isn't connected to the DHT network.
  - **DHT: X nodes** (where X is an integer) means DHT is enabled, and  $\mu$ Torrent is directly connected to X other clients on the DHT network. The keyword is "directly," as this number is not an indication of the number of users you are effectively connected to on the DHT network -- that number is much greater.

- 
- The **Network Status** section displays an icon to signify your connectability to people outside of your firewall (software and/or hardware). **Left-click**ing on the icon will open the **Setup Guide**. Note that the network status light can be blank as well, in which case it simply means that no torrent jobs are started, so  $\mu$ Torrent is unable to check the port status.

-  means everything is fine, and you are receiving incoming connections.
-  means no incoming connections have been received yet. Seeing this icon doesn't automatically mean something is wrong unless it remains yellow for an extended period of time while there are actively transferring torrent jobs. If you're unsure about whether your port is forwarded properly, then the best course of action is to use the port checker in the Setup Guide.
-  means  $\mu$ Torrent wasn't able to use the listening port. Generally, this means a firewall is blocking the port (in which case you should **attempt to fix it**), but it's also possible that another application is using this port, in which case  $\mu$ Torrent will inform you that there is a listening error.

- 
- The **Download** section starts with a "**D**" and displays download speed related statistics in the following order:

- The global download rate limit (in square brackets, assuming such a limit is set, and **Show speed limits in the status bar** is enabled)
- The total data download speed
- The total communication download overhead speed (with an "**O**" preceding the speed)
- The total local peer download speed (with an "**L**" preceding the speed, assuming  $\mu$ Torrent is downloading at more than 1 KiB/s from local peers)
- The total amount of data downloaded this session (with a "**T**" preceding the amount)

- 
- The **Upload** section starts with a "**U**" and displays upload speed related statistics in the following order:

- The global upload rate limit (in square brackets, assuming such a limit is set, and **Show speed limits in the status bar** is enabled)
- The total data upload speed
- The total communication upload overhead speed (with an "**O**" preceding the speed)
- The total local peer upload speed (with an "**L**" preceding the speed, assuming  $\mu$ Torrent is uploading at more than 1 KiB/s to local peers)
- The total amount of data uploaded this session (with a "**T**" preceding the amount)

- The **Links** section contains optional icon-links to µTorrent's Facebook's web-page and Tweeter's web-page. Both links are one-time only links, and will be permanently removed once you click them.

## Context Menu

Depending on where you **right-click** the status bar, you will get different menus to control µTorrent's behavior.

- **Right-click**ing anywhere to the left of the Download section will show a context menu with the following options:
  - **Enable Scheduler** allows you to enable or disable the scheduler from the without having to **toggle it manually** in the Preferences.
  - **Enable DHT** allows you to enable or disable DHT from the without having to **toggle it manually** in the Preferences.
- **Right-click**ing on the **Download** section will show a context menu that allows you to control the global maximum download limit. The values displayed depend on **the speed popup list settings** in the Preferences.
- **Right-click**ing on the **Upload** section will show a context menu that allows you to control the global maximum upload limit. The values displayed depend on **the speed popup list settings** in the Preferences.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface](#)

# Preferences

This section provides descriptions of all options available in the  $\mu$ Torrent Preferences dialog. Nearly all of  $\mu$ Torrent's internal settings can be configured here. To access the Preferences, select "*Options*" then "*Preferences...*" in the **main menu**, or press **Ctrl + P** on the keyboard. The last selected section before the Preferences dialog is closed will be automatically selected the next time the user enters the Preferences.

- General
- UI Settings
- Directories
- Connection
- Bandwidth
- BitTorrent
- Transfer Cap
- Queueing
- Scheduler
- Remote
- Playback
- Paired Devices
- Advanced
  - UI Extras
  - Disk Cache
  - Web UI
  - Run Program

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

# General

## Language

- The **Language** dropdown menu allows you to select the language for the entire µTorrent interface. In order for this to work, you must have installed the language pack ([utorrent.lng](#)) into µTorrent's settings directory, or in the same directory as the µTorrent executable itself.
- The **More...** button opens the [download page](#) in your web browser, where you'll find the language pack available for download.

## Windows Integration

- The **Associate with torrent files** button tells Windows to open .torrent files, .btsearch files and magnet URIs in µTorrent. If [maindoc.ico](#) is present in µTorrent's settings directory, it will be used as the icon for .torrent files instead of making Windows attempt to generate an icon automatically.
- The **Install IPv6/Teredo** button makes µTorrent install IPv6/Teredo support on the computer. Note that this option works only on operating systems released after Windows XP with at least Service Pack 2 (SP2) installed.
- **Check association on startup** will tell µTorrent to check associations with anything it is associated with each time it is run. If the association no longer belongs to µTorrent at start, the user will be prompted for further action.
- **Start µTorrent when Windows starts** will make µTorrent start when Windows starts up.
- The **Start minimized** will make µTorrent start minimized when Windows starts.

## Updates and Privacy

- **Automatically install updates** will automatically install new versions of  $\mu$ Torrent when they become available. The check for new versions is done on startup and periodically thereafter, waiting 24 hours between each update check (automatic or manual). If unchecked, you will need to update  $\mu$ Torrent manually.
- **Update to beta versions** will update  $\mu$ Torrent to beta versions if checked. If unchecked,  $\mu$ Torrent will only upgrade to stable release builds.
- **Notify me before installing updates** will ask the user to confirm updates to  $\mu$ Torrent if checked. If unchecked,  $\mu$ Torrent will automatically update.
- **Send detailed info when checking for updates (always anonymous)** allows  $\mu$ Torrent to send a randomly generated ID and the  $\mu$ Torrent version and build numbers to the update server during update checks in order to count the number of people there are using  $\mu$ Torrent. Whether or not this option is enabled, all communication with the  $\mu$ Torrent servers are always anonymous.
- The **Use Boss-Key password** field allows you to select a keyboard combination that, when pressed, will toggle the visibility of all traces  $\mu$ Torrent on the immediate desktop. The main window and any open dialogs, the taskbar entry, and the system tray icon all disappear when hidden, and appear again when unhidden (by pressing the keyboard combination a second time). To set this field, you must press the keyboard combination you wish to set the boss-key to. To do so, press your choice of modifier keys (any combination of **Ctrl**, **Alt**, and **Shift**) and the key (letters, numbers, function keys, and so on). Make sure the key combination you select is not used by another application, as that can lead to conflicts.
- The **Clear Private Data** button clears previously used paths in various locations, such as the **Add New Torrent** dialog and the **Create New Torrent** dialog.

## When Downloading

- **Append .!ut to incomplete files** tells  $\mu$ Torrent to append the .!ut extension to any file in the torrent contents that still hasn't finished downloading. Once a file is completed, the extension gets removed from it. Note that the state of this option takes effect immediately after the preferences are saved, regardless of whether torrent jobs are started, stopped, paused, or queued.
- **Pre-allocate all files** tells  $\mu$ Torrent to create and fully allocate every file you select to download immediately after starting the torrent job. Note that this option does not have an impact on hard drive fragmentation (advantageous or otherwise), as  $\mu$ Torrent already allocates each file upon writing to disk even without this option.
- **Prevent standby if there are active torrents** is self-explanatory; if there are active torrent jobs,  $\mu$ Torrent will attempt to keep the computer out of automatic standby mode.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

# UI Settings

## Display Options

- **Confirm when deleting torrents** displays a dialog asking for confirmation when you try to delete a torrent job from [the torrent jobs list](#).
- **Confirm when deleting trackers** displays a dialog asking for confirmation when you try to delete a tracker from [the Trackers list](#).
- **Show confirmation dialog on exit** displays a dialog asking for confirmation when you try to exit  $\mu$ Torrent.
- **Alternate list background color** will make  $\mu$ Torrent alternate the background color for each item in a list-view (be it the [torrent jobs list](#), or one of the list-views in the [detailed info pane](#)) between the default list-view background color and the column highlight color. If disabled,  $\mu$ Torrent will simply display the default list-view background behind every item in the list-view.
- **Show current speed in the title bar** displays the total upload and download rates in the title bar of the main window, before " $\mu$ Torrent".
- **Show speed limits in the status bar** displays the global rate limit (if set) next to its corresponding value in the status bar.
- **Use fine grained file properties** displays 16 priority levels in [the files tab](#) instead of the regular three (normal/high/low).
- **Confirm exit if critical seeder** displays a confirmation dialogue when you try to exit and you are a critical-only seeder

## System Tray

- **Close button closes uT to tray** will tell  $\mu$ Torrent to close the main

window to system tray rather than exit when the close button is used in the window title bar.

- **Minimize button minimizes uT to tray** will tell  $\mu$ Torrent to minimize the main window to system tray rather than the taskbar.
- **Always show tray icon** will display the tray icon regardless of  $\mu$ Torrent's main window visibility (unless hidden with the **boss-key**). If disabled,  $\mu$ Torrent will only display the tray icon when it is closed or minimized to system tray.
- **Single click on tray icon to open** will make it so that it takes a single **left-click** on the tray icon to display or hide the main window. If left unchecked, it would require a **double-click** to achieve the same result.
- **Show balloon notifications in tray** will display a bubble tooltip notification in the system tray when certain events occur in  $\mu$ Torrent, such as the download completion of a torrent job, an error, and so on.
- **Always activate when clicked** will make it so that **left-click**ing (or **double-click**ing, depending on whether the single click tray icon option is enabled or disabled) the tray icon will only activate the main window. If left disabled,  $\mu$ Torrent would normally alternate between activating and hiding the window for the same action.

## When Adding Torrents

- **Don't start the download automatically** adds manually imported torrent jobs in stopped mode. If this option is enabled, "*Start torrent*" in the **Add New Torrent** dialog will be unchecked by default (but can still be enabled). If that dialog is not enabled, torrent jobs will simply be added in stopped mode, and will have to be started manually.
- **Activate the program window** will open the  $\mu$ Torrent main window (if hidden, minimized to system tray, or minimized to taskbar). If unchecked,  $\mu$ Torrent will simply display the **Add New Torrent** dialog.

- **Show a window that displays the files inside the torrent** will tell  $\mu$ Torrent to display the **Add New Torrent** dialog. If unchecked,  $\mu$ Torrent will simply ask where you'd like to save the torrent contents, assuming a default download location is not set.

## Actions for Double Click

- **For seeding torrents** allows you to select the action that  $\mu$ Torrent takes when you `double-click` a seeding torrent job on the torrent jobs list.
- **For downloading torrents** allows you to select the action that  $\mu$ Torrent takes when you `double-click` a downloading torrent job on the torrent jobs list.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Preferences](#)

# Directories

## Location of Downloaded Files

- **Put new downloads in** specifies the default location that the contents of newly added torrent jobs will be downloaded to.
  - **Always show dialog on manual add** tells  $\mu$ Torrent to display the [Add New Torrent](#) dialog even if a default download location is. If left unchecked,  $\mu$ Torrent automatically sets the download location to the specified location and adds the torrent job to the torrent jobs list accordingly, without user intervention.
- **Move completed downloads to** specifies the location that torrent contents that have finished downloading should be moved to. Files are moved immediately after the torrent job finishes downloading and the files are written to disk.
  - **Append the torrent's label** will add the torrent job's label to the path that the torrent contents are moved to after they finish downloading.
  - **Only move from the default download directory** tells  $\mu$ Torrent to move torrent contents only if it was saved to the directory specified under "*Put new downloads in*". If that option wasn't set, this option should be left disabled.

## Location of .torrents

- **Store .torrents in** sets the location that  $\mu$ Torrent should save .torrent files to. By default,  $\mu$ Torrent will save these files to the [settings directory](#).
- **Move .torrents for finished jobs to** moves .torrent files to the specified location after it is completed. By default, these files remain in the same location that they were first copied to.
- **Automatically load .torrents in directory** tells  $\mu$ Torrent to check the specified directory every second for new new .torrent files, and automatically load them. Note that the directory specified here must be different from the directories specified in the .torrent storage settings, otherwise, you will run into problems with .torrent files looking like they're missing to  $\mu$ Torrent.

- **Delete loaded .torrents** tells μTorrent to delete the .torrent file when it is auto-loaded. Normally, if this option is not checked, μTorrent appends .loaded to the extension to indicate that it has been loaded and to prevent μTorrent from trying to load the .torrent file again.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

# Connection

## Listening Port

- The **Port used for incoming connections** field specifies the port that  $\mu$ Torrent should use to listen for incoming connections. This port must be **unfirewalled** for optimal usage of  $\mu$ Torrent. Setting the port to 0 indicates to  $\mu$ Torrent that you would like for it to select a random port after the changes are confirmed.
- The **Random port** button selects a random port between 10000 and 65000 as the listening port.
- **Enable UPnP port mapping** allows  $\mu$ Torrent to communicate with the router to forward a port without your manual intervention. Some devices do not support Universal Plug and Play (UPnP), so you might still have to forward your ports manually. Disable UPnP if that is the case.
- **Enable NAT-PMP port mapping** allows  $\mu$ Torrent to attempt to forward a port with routers that support the NAT port mapping protocol (Apple products, for example).
- **Randomize port each start** makes  $\mu$ Torrent select a random port between 1000 and 65000 to use at startup. Note that this option might conflict with your port forwarding settings, so if you manually open ports in your router and firewall, it's best to leave this option disabled.
- **Add Windows Firewall exception** allows  $\mu$ Torrent to add an entry to the Windows Firewall exceptions list that lets it bypass the firewall. This is useful only if you actually have Windows Firewall enabled. Note that this option works only on operating systems released after (and including) Windows XP with at least Service Pack 2 (SP2) installed.

## Proxy Server

- The **Type** dropdown menu specifies the type of proxy you are trying to use.
- The **Proxy** field is where you enter the proxy's URL or IP address.
- The **Port** field is where you enter the proxy's port.
- Checking **Authentication** indicates that you need to log into the proxy server in order for it to work.
  - The **Username** field is where you enter the username required to log into the proxy server for authentication.
  - The **Password** field is where you enter the password required to log into the proxy server for authentication.
- **Use proxy for hostname lookups** forces  $\mu$ Torrent to perform hostname (DNS) lookups through the proxy. Reverse DNS is not proxied.
- **Use proxy for peer-to-peer connections** forces  $\mu$ Torrent to communicate and transfer data with peers through the proxy. By default, this option is disabled, and  $\mu$ Torrent only uses the proxy to communicate with trackers. This option may not work with some HTTP proxies (not all HTTP proxies support HTTP CONNECT).
- **Note:**  $\mu$ Torrent can only proxy UDP-based communication through SOCKS5 proxies. Proxying of IPv6 traffic is currently not supported. By default, connection types that your proxy does not support will simply connect as normal.

## Proxy Privacy

- **Disable all local DNS lookups** will disable reverse DNS and prevent hostname lookups from bypassing the proxy. This will not function properly without enabling "*Use proxy for hostname lookups.*" If your proxy is misconfigured or not working, then DNS will simply fail. As a safeguard,  $\mu$ Torrent's autoupdates and crash reports will bypass this option.

- **Disable features that leak identifying information** will prevent  $\mu$ Torrent from sharing your non-proxied IP through handshakes with other peers, as well as through DHT. It will also prevent it from handing out your IPv6 address to IPv4 peers and vice versa.
- **Disable connections unsupported by the proxy** will disable connection types that your proxy type cannot handle. For HTTP, HTTPS and SOCKS4 proxies, this will disable all UDP-based communication (DHT, uTP, UDP trackers, IPv6). For SOCKS5, it will only disable IPv6, as IPv6 is currently not proxied. If your proxy is misconfigured or not working, all traffic in the client will stop.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

# Bandwidth

## Global Upload Rate Limiting

- The **Maximum upload rate** field limits the maximum rate at which  $\mu$ Torrent will upload. Setting the value to 0 is equivalent to setting it to unlimited. This value is interpreted in KiB/s, so please enter it as such.
- The **Alternate upload rate when not downloading** field allows you to define an alternate upload rate when no torrent job is in downloading mode (only seeding). Other than the context in which it is active, this upload rate limit behaves exactly like the "*Maximum upload rate*" field. Unless you understand well the consequences of enabling this option, refrain from using it, as it might give the impression that  $\mu$ Torrent is not obeying the global maximum upload rate. This value is interpreted in KiB/s, so please enter it as such.

## Global Download Rate Limiting

- The **Maximum download rate** field limits the maximum rate at which  $\mu$ Torrent will download. Setting the value to 0 is equivalent to setting it to unlimited. Note that download rate limiting is very unreliable, so you might see greater fluctuations in download speeds with this option set to anything other than 0, and  $\mu$ Torrent might not be able to limit the speeds to the specified amount either. This value is interpreted in KiB/s, so please enter it as such.

## Number of Connections

- The **Global maximum number of connections** field defines the maximum number of connections that  $\mu$ Torrent can make at any given time in all active torrent jobs.
- The **Maximum number of connected peers per torrent** field

defines the maximum number of peers that  $\mu$ Torrent can connect to at any given moment for each torrent job.

- The **Number of upload slots per torrent** field defines the maximum number of peers that  $\mu$ Torrent will upload to at any given moment for each torrent job.
- **Use additional upload slots if upload speed < 90%** tells  $\mu$ Torrent to open more upload slots if the current total upload speed is less than 90% of the **global maximum upload rate**.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

# BitTorrent

## Basic BitTorrent Features

- **Enable DHT Network** enables the Distributed Hash Table (DHT) if checked. You can also do this by **right-click**ing the DHT status in the main window status bar and selecting the corresponding option.
- **Enable DHT for new torrents** tells  $\mu$ Torrent to check for peers from the DHT network on each newly added torrent job. This option only works if DHT is enabled, and if the 'private' flag is not set in the .torrent file.
- **Enable Local Peer Discovery** allows  $\mu$ Torrent to attempt to discover local peers via multicast. If a peer is considered local, transfer rates with the peer will not be limited unless **Limit local peer bandwidth** is also enabled. Local Peer Discovery will not be used for .torrent files with the 'private' flag set. This option serves as a default setting for newly-added torrent jobs, and does not retroactively affect previously-added. This option can be disabled on a per torrent job basis in the respective torrent job's **Properties** dialog.
- **Enable bandwidth management** allows  $\mu$ Torrent to use uTP to try and gain a better sense of how much bandwidth it should be using while maintaining quality of Internet service.
- **Enable UDP tracker support** allows  $\mu$ Torrent to communicate with trackers using the UDP-based tracker communication protocol. Such trackers have URLs that begin with "*udp://*".
- **Ask tracker for scrape information** tells  $\mu$ Torrent to scrape the tracker (ask for peer and seed counts). By default, the tracker is scraped only when the torrent job is started, but **this can be changed**.
- **Enable Peer Exchange** enables Peer Exchange (PEX) for each

newly added torrent job, provided the 'private' flag is not set for that .torrent file. This option serves as a default setting for newly-added torrent jobs, and does not retroactively affect previously-added. This option can be disabled on a per torrent job basis in the respective torrent job's **Properties** dialog.

- **Limit local peer bandwidth:** This option decides whether  $\mu$ Torrent should apply the rate limits to traffic between itself and peers on the local network. Peers are considered local if they are discovered by Local Peer Discovery, or if they are on the same LAN as the client.
- The **IP/Hostname to report to tracker** field allows you to specify your IP address when reporting to the tracker. This is used in the case where your WAN (Internet) IP address is not reported correctly to the tracker for whatever reason (might be that you are behind a proxy server). Note that you can type a hostname in this field, so if you use a dynamic DNS service, you can enter your domain into this field. Also, realize that not all trackers pay attention to this information, and often ignore it.

## Protocol Encryption

- The **Outgoing** dropdown menu allows you to select the mode of encryption that you prefer  $\mu$ Torrent to establish. All modes will accept incoming encrypted connections, and the encryption is 2-way.
  - **Disabled** will force  $\mu$ Torrent to attempt to establish only unencrypted outgoing connections.
  - **Enabled** will allow  $\mu$ Torrent to establish encrypted and unencrypted outgoing connections, depending on how the peer responds to the handshake. This option provides  $\mu$ Torrent with the largest pool of peers to pick from for connecting to, but incurs additional overhead that may often be unnecessary in most typical situations.
  - **Forced** forces  $\mu$ Torrent to establish only encrypted outgoing connections. Any peer that doesn't support encryption will not be connected to. It is recommended that you not use this option unless your ISP actively searches for unencrypted outgoing connections, as it can impair your ability to connect to peers.
- **Allow incoming legacy connections** allows  $\mu$ Torrent to accept unencrypted incoming connections. If disabled, any incoming

connection that is unencrypted will be ignored. It is recommended that you not disable this option unless your ISP actively searches for unencrypted incoming connections, as it can significantly impair your ability to connect to peers.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

# Transfer Cap

The transfer cap is used to control the amount of data  $\mu$ Torrent will transfer over a specified period of time.

**Enable Transfer Cap** will enable the transfer cap if checked.

## Cap Settings

- The **Bandwidth Cap** field sets the amount of data to use as the transfer cap for the selected period.
- The **Time Period** dropdown menu sets the number of days to use as the transfer cap period.
- The **Limit Type** option sets the transfer direction(s) to count as part of the transfer cap.

## Usage History

This displays textual statistics pertaining to the amount of data transferred for the selected time period. The "*Reset History*" button below resets the collected statistics.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

# Queueing

## Queue Settings

- The **Maximum number of active torrents** field defines the maximum number of unforced torrent jobs that  $\mu$ Torrent will allow to be active before placing it on the queue. Torrent jobs are counted regardless of whether they're seeding or downloading torrents, as long as they are uploading at rates above the value specified by `queue.slow_ul_threshold` or downloading at rates above the value specified by `queue.slow_dl_threshold`.
- The **Maximum number of active downloads** field defines the maximum number of unforced torrent jobs that  $\mu$ Torrent will allow to be downloaded before making it wait on the download queue. This option only applies to torrent jobs that are downloading or are to be placed in downloading mode.

## Seeding Goal

- The **Minimum ratio** field allows you to set the ratio that you wish to reach before  $\mu$ Torrent throttles the speed for the torrent job (or stops it, if you set it to do so). Setting the ratio to -1 is equivalent to setting it to unlimited. Setting this value to 0 tells  $\mu$ Torrent to ignore this value and look only at the seeding time limit. This value is interpreted as a percentage.  $\mu$ Torrent will throttle the seeding process only after both this and time limit have been reached.
- The **Minimum seeding time** field allows you to specify the minimum amount of time you wish for the torrent job to continue seeding at normal speeds after it has finished downloading.  $\mu$ Torrent will throttle the seeding process only after both this and the ratio threshold have been reached. This value is interpreted in minutes.
- **Seeding tasks have higher priority than downloading tasks** will

give seeding tasks higher priority than downloads, so if your maximum number of active torrents is reached, and a torrent job reaches seeding state, the downloading tasks will not force it into queued seeding state.

- **Minimum number of available seeds** will keep seeding till the specified number of seeds are available for this torrent
- **Note:** These values only affect torrent jobs added after they are set. Existing torrent jobs will retain their current seeding goals, even if these default settings are modified.

## When $\mu$ Torrent Reaches the Seeding Goal

- The **Limit the upload rate to** field allows you to set the speed that  $\mu$ Torrent throttles the upload speed for a torrent job to when it reaches the seeding goal set. Setting this value to 0 is equivalent to telling  $\mu$ Torrent to stop the torrent job. A change to this value affects only torrent jobs that have not yet reached their seeding goals. This value is interpreted in KiB/s, so please enter it as such.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

# Scheduler

The scheduler is used to control  $\mu$ Torrent during certain times of the day on certain days of the week. With the scheduler, it is possible to tell  $\mu$ Torrent to stop torrent jobs, use a different bandwidth limit, or stop downloading entirely during specified times.

**Enable Scheduler** will enable the scheduler if checked.

## Scheduler Table

- Each square in this table represents each hour in each day of each week. You can change the color by **left-click**ing on a box. It cycles through dark green, light green, white, and light red. **Left-click-drag**ging will cause all time blocks that the mouse passes over to change to the new color.
- **Full Speed** occurs during the time periods where the scheduler table is colored dark green. During these time slots,  $\mu$ Torrent will adhere to the normal **global upload** and **global download** rate limits.
- **Limited** occurs during the time periods where the scheduler table is colored light green. During these time slots,  $\mu$ Torrent will adhere to the upload and download rate limits specified in the **Scheduler Settings**. Because the settings affect global rate limits, forced torrent jobs are affected during these time slots.
- **Turn off** occurs during the time periods where the scheduler table is colored white. During these time slots,  $\mu$ Torrent will stop any unforced torrent jobs.
- **Seeding only** occurs during the time periods where the scheduler table is colored light red. During these time slots,  $\mu$ Torrent will cause unforced torrent jobs to stop downloading, though they will continue to upload data. Forced torrent jobs will continue downloading during these time slots.

## Scheduler Settings

- The **Limited upload rate** field sets the global maximum upload rate that all torrent jobs should follow when operating under Limited mode. Other than the context in which it is active, this upload rate limit behaves exactly like the **global maximum upload rate**. Leaving the field blank is equivalent to setting it to unlimited. This value is interpreted in KiB/s, so please enter it as such.
- The **Limited download rate** field sets the global maximum download rate that all torrent jobs should follow when operating under Limited mode. Other than the context in which it is active, this download rate limit behaves exactly like the **global maximum download rate**. Leaving the field blank is equivalent to setting it to unlimited. This value is interpreted in KiB/s, so please enter it as such.
- **Disable DHT when turning off** disables DHT when the scheduler is operating under Turn off mode.

[« Previous](#)   [Next »](#)

[µTorrent User Manual > Appendix A: The µTorrent Interface >](#)

[Preferences](#)

# Remote

**uTorrent Remote** provides an easy and highly secure way to access your client remotely using a browser. Detailed instructions and opening an account are provided [in here](#), examples and guides are available [here as well](#).

**Enable uTorrent remote access** will enable the remote access to uTorrent if checked. Once enabled - you will be able to observe the status to the remote access server. Logging in remotely is though here: <https://remote.utorrent.com/>

## Authentication

- **Computer name:** enter the account/user/computer name you have opened for remote access.
- **Password:** enter the password you have selected for your account/access.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Preferences](#)

# Playback

Playback allows you to seamlessly play your audio/video files with your default player based on file extension or specify a specific player to use for all files.

## Playback Player Selection

- **Choose your default player** allows you to specify which player to be used for all files for playback. You can select one of the automatically setup players in the list or browse for your own player. Use %1 to specify the filename parameter in the command line (Required).

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >

[Preferences](#)

## Paired Devices

The paired devices dialogue will show you the currently available devices that you can stream data to or copy converted files to their USB storage.

### Paired Devices List

- **Forget Faired Device:** You can select a device and press to forget/erase any device from the list.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Preferences](#)

# Advanced

- UI Extras
- Disk Cache
- Web UI
- Run Program

**WARNING:** Unless you fully understand the consequences of changing an advanced setting, you should not modify it without guidance.

## Notes:

**3.3:** This option is applicable only from µTorrent version 3.3 and above

- 
- **bt.allow\_same\_ip:** Enabling this option allows multiple incoming connections from the same IP address. This option affects a single torrent job at a time, so you can still have the same IP address connect to you on different torrent swarms. It is recommended that this option be left disabled, as it weakens the anti-leech protection.
  - **bt.ban\_ratio:** The lowest acceptable ratio of good to bad pieces a peer can send before it gets banned. The lower this option is set, the more forgiving µTorrent will be toward bad pieces, meaning that it will be less likely to ban a peer. This takes effect after **bt.ban\_threshold** is exceeded and **bt.use\_ban\_ratio** is enabled.
  - **bt.ban\_threshold:** This option specifies the maximum number of hashfailed pieces any single peer can send before µTorrent takes action against it (either banning it outright, or enforcing **bt.ban\_ratio** if **bt.use\_ban\_ratio** is enabled).
  - **bt.compact\_allocation:** Enabling this option allows µTorrent to create files in a manner such that the data are incrementally written

to disk without file pre-allocation. Because writes are compact, enabling this option may potentially lead to an increased level of disk fragmentation while the file remains incomplete. In addition, this option further decreases the already-low probability that a file can be previewed before completion, since it may write the data for in-progress files out of order. Here are some things to take note of when using this option:

- If you tell  $\mu$ Torrent to **pre-allocate all disk space**, this option is ignored, and  $\mu$ Torrent will pre-allocate the file anyway.
  - If this option is enabled, files **can't be skipped**. If a torrent job has skipped files, it will not use compact allocation.
- **bt.connect\_speed**: This option specifies the number of connections  $\mu$ Torrent should allow to be attempted and/or established each second, whether the connections use uTP or TCP.
  - **bt.determine\_encoded\_rate\_for\_streamables**: Enabling this option shows media files' rate in Files tab
  - **bt.enable\_pulse**: Enabling this option effects ratings, and disables comments too.
  - **bt.enable\_tracker**: Enabling this option enables the rudimentary tracker embedded in  $\mu$ Torrent. If you wish to use this tracker, the URL is located at `http://IP:port/announce`, where IP is your WAN IP address, and port is **the port  $\mu$ Torrent is listening on** (or the **alternative listening port** if set and enabled). If you use a dynamic DNS service, your domain may be used instead of your IP address. The embedded tracker allows tracking of external .torrent files, and provides no way to limit them. There is no interface for viewing the .torrent files that are tracked. It is imperative that  $\mu$ Torrent is able to listen for incoming connections for this feature to work properly, so you have to make sure you have completely forwarded your ports in order to use the embedded tracker.
  - **bt.graceful\_shutdown**: If enabled,  $\mu$ Torrent will take as long as it needs to finish its shutdown sequence (writing in-progress pieces to disk, deleting files in deletion queue, and waiting for tracker replies

to stop messages -- among other things). That means that even if it takes several minutes to shutdown gracefully, it will wait for that long, and the process will remain in memory until then. If disabled,  $\mu$ Torrent will limit how long it waits to to 10 seconds, and regardless of the state of the shutdown sequence,  $\mu$ Torrent will force itself to exit.

- **bt.multiscrape:** Enabling this option allows  $\mu$ Torrent to send multiple hashes each time it scrapes a tracker, which is more efficient than sending one hash at a time. In most circumstances, this option should not need to be disabled, as  $\mu$ Torrent will fall back to single scraping if it detects that the tracker does not support multi-scraping.
- **bt.no\_connect\_to\_services:** This option tells  $\mu$ Torrent not to connect to peers using ports specified in [bt.no\\_connect\\_to\\_services\\_list](#) as their listening ports. This stops firewalls from complaining about  $\mu$ Torrent trying to send an e-mail.
- **bt.no\_connect\_to\_services\_list:** This option specifies which ports  $\mu$ Torrent should not connect to when [bt.no\\_connect\\_to\\_services](#) is enabled.
- **bt.prio\_first\_last\_piece:** Enabling this option prioritizes the first and last pieces of each file in a torrent job, increasing the chances that they can be previewed before download completion.  $\mu$ Torrent will prioritize at least the first and last 1 MiB of data in a file.
- **bt.prioritize\_partial\_pieces:** Enabling this option ('True'), tells  $\mu$ Torrent to always try to request blocks from pieces we've already started.
- **bt.randomize\_peer\_id:** Disabling this option tells  $\mu$ Torrent not to change peerid on every connection. It still uses unique one for tracker, and should also be random limit. This randomization is not being used for private torrents.
- **bt.ratelimit\_tcp\_only:** Enabling this option tells  $\mu$ Torrent to limit the

upload and download rates for TCP connections based on information received over the uTP transport rather than using static global rate limits. This option is ignored if `bt.tcp_rate_control` is disabled.

- **bt.rcv\_max\_sockbuf:** This debugging option defines  $\mu$ Torrent's local receive buffer socket size. Tweaking can reduce memory usage, but may hurt performance if too low. [3.3](#)
- **bt.read\_only\_on\_complete:** This option makes downloaded files have the read-only attribute set when torrent is complete. This prevents MP3s and the like from being retagged by media players and corrupted. [3.3](#)
- **bt.save\_resume\_rate:** This option set the time interval for saving the "resume" data to every n seconds.
- **bt.scrape\_stopped:** Enabling this option allows  $\mu$ Torrent to get seed and peer counts for torrent jobs that are stopped.
- **bt.send\_have\_to\_seed:** Enabling this option tells  $\mu$ Torrent to send a message to other seeds indicating how many pieces you currently have.
- **bt.set\_sockbuf:** This debugging option allows  $\mu$ Torrent to automatically detect the TCP buffer size periodically (`so_sndbuf`) and adjust it based on your upload speed. It does not adjust based on latency.
- **bt.shutdown\_tracker\_timeout:** This option controls the maximum amount of time  $\mu$ Torrent will wait, when exiting, for each tracker to respond to a stopped event before it forces itself to terminate. This value is interpreted in seconds, so please enter it as such. Setting this value to 0 tells  $\mu$ Torrent to wait for an indefinite amount of time until it receives a response.
- **bt.shutdown\_upnp\_timeout:** This option controls the maximum amount of time  $\mu$ Torrent will wait, when exiting, for routers to

respond to a request to un-map the listening ports before it forces itself to terminate. This value is interpreted in seconds, so please enter it as such. Setting this value to 0 tells  $\mu$ Torrent to wait for an indefinite amount of time until it receives a response.

- **bt.send\_max\_sockbuf**: This debugging option defines  $\mu$ Torrent's local send buffer socket size. Tweaking can reduce memory usage, but may hurt performance if too low. [3.3](#)
- **bt.tcp\_rate\_control**: Enabling this option tells  $\mu$ Torrent to use information from the uTP transport as hints for limiting TCP transfer rates.
- **bt.transp\_disposition**: This option controls  $\mu$ Torrent's level of bias towards using TCP or uTP for transporting data (assuming the peer at the other end of the connection supports both transport protocols). The following is a list of the accepted values:
  - **1** allows  $\mu$ Torrent to attempt outgoing TCP connections
  - **2** allows  $\mu$ Torrent to attempt outgoing uTP connections
  - **4** allows  $\mu$ Torrent to accept incoming TCP connections
  - **8** allows  $\mu$ Torrent to accept incoming uTP connections
  - **16** tells  $\mu$ Torrent to use the new uTP header. This is an improved communication header, but is not backwards compatible with clients that do not understand it.

This option is interpreted as a bitfield, so values can be added together to obtain a combination of behaviors. Setting this value to 255 guarantees that all behaviors are enabled.

- **bt.use\_ban\_ratio**: This option tells  $\mu$ Torrent to use [bt.ban\\_ratio](#) to decide when a peer gets banned after it has exceeded [bt.ban\\_threshold](#).
- **bt.use\_dns\_tracker\_prefs**: This option tells  $\mu$ Torrent to respect or ignore [BEP34](#).
- **bt.use\_rangeblock**: When enabled,  $\mu$ Torrent will automatically attempt to determine whether an entire range of IP addresses should be banned for sending hashfailed pieces rather than banning

individual IPs one at a time. When  $\mu$ Torrent bans 4 IPs from the same /24 CIDR block, it will ban the entire /24 CIDR block. When  $\mu$ Torrent bans 4 CIDR blocks of size /24 from the same /16 CIDR block, it will ban the entire /16 CIDR block. When  $\mu$ Torrent bans 4 CIDR blocks of size /16 from the same /8 CIDR block, it will ban the entire /8 CIDR block.

- **diskio.all\_writes\_sync**: Enabling this option causes  $\mu$ Torrent to open files in synchronous mode so all writes are immediately flushed to disk [3.3](#)
- **diskio.cache\_reduce\_minutes**: This option determine how often (in minutes)  $\mu$ Torrent compacts the disk cache
- **diskio.cache\_stripe**: This option tells  $\mu$ Torrent the size of blocks of memory used in it's disk cache in KiB. Minimum of piece size and cache stripe are in KiB.
- **diskio.coalesce\_write\_size**: This option determines the size threshold for which  $\mu$ Torrent should write data out coalesced, and is relevant only if [diskio.coalesce\\_writes](#) is enabled. This value is interpreted in bytes per second, so please enter it as such.
- **diskio.coalesce\_writes**: This option tells  $\mu$ Torrent to try to minimize the number of writes to disk by writing more data at once. It doesn't have any effect on download speeds, but might increase memory and CPU usage to achieve less disk writes.
- **diskio.flush\_files**: Enabling this option causes  $\mu$ Torrent to close file handles every minute. It helps to reduce the effect of Windows managing the system cache badly for some people and causing apparent "memory leaks."
- **diskio.mark\_of\_the\_web**: This option sets NTFS ADS that tells Windows that this file was downloaded from the Internet [3.3](#)
- **diskio.max\_write\_queue**: This option sets the maximum depth of

the write queue before the client starts showing disk overloaded [3.3](#)

- **diskio.minimize\_kernel\_caching** : This option disables compact allocation, might be POSIX only [3.3](#)
- **diskio.no\_zero**: Enabling this option causes  $\mu$ Torrent to skip the zero-filling process for file allocation. This option works only on Windows XP or newer, and requires administrator privileges by default. However, it is possible to make this work on limited accounts by setting the "Perform volume maintenance tasks" policy appropriately in the Windows Group Policy Editor. Skipping zero-filling speeds up the file allocation process, but because the allocated files have shared read access, there is a risk that any sensitive data that may have once existed at that location in disk but isn't wiped will potentially be exposed for other applications and users to read, including those without volume maintenance privileges.
- **diskio.quick\_hash**: Toggles advanced optimization when verifying a torrent data's integrity.
- **diskio.resume\_min**: Megabytes to be free on disk before torrent resumes
- **diskio.smart\_hash**: This option makes  $\mu$ Torrent hash data from memory (if in the write queue) instead of flushing to disk, re-reading from disk, and then hashing. This should help reduce hard disk reads, especially when transferring at high speeds.
- **diskio.smart\_sparse\_hash**: This option is a workaround for a problem in some versions of Windows that return incorrect data to  $\mu$ Torrent regarding sparse files and the amount of data that has actually been completed on disk.
- **diskio.sparse\_files**: Enabling this option causes  $\mu$ Torrent to allocate only the data that it writes, but will inform the filesystem of the file's size (so that it can attempt to reserve enough contiguous space on the hard drive without having to physically zero all of the

space out for the file). Even though space is reserved for the file, no space will be taken for the unwritten parts of the file. Enabling this option may potentially lead to increased disk fragmentation in rare cases where the drive does not have enough free space available to honor the space reservation for sparse files. Here are some things to take note of when using this option:

- Sparse files work only on partitions that are formatted as NTFS.
  - Hash checking sparse files tends to be quicker than hash checking pre-allocated files, as  $\mu$ Torrent won't have to hash zeroed-out pre-allocated data.
  - On Windows Vista, sparse files can cause  $\mu$ Torrent to run into a [file system limitation](#).
  - If you are using a non-administrator account with a disk quota, sparse files won't work, and the file will still get fully allocated. This is a limitation with Windows that  $\mu$ Torrent can't do anything about.
  - This option cannot be used in conjunction with [pre-allocate all files](#). If both options are enabled simultaneously, pre-allocation will take precedence.
  - When used in conjunction with [bt.compact\\_allocation](#),  $\mu$ Torrent will reserve space for each file in the filesystem, but it will continue to use compact writes.
- **diskio.use\_partfile:** This option is used to store data that is downloaded from files that you told  $\mu$ Torrent to skip. This is necessary to prevent the file from being allocated. It separately stores the parts of the skipped files that come with a piece, since  $\mu$ Torrent must download and save the entire piece in order to confirm that it is uncorrupted, and each piece can contain data from multiple files. The partfile is removed when you remove the torrent job from the torrent job list.
  - **distributed\_share.enable:** This option enables the participation in distributed backups [3.3](#)
  - **gui.auto\_restart:** This option configures crash recovery in  $\mu$ Torrent. When this option is enabled, if  $\mu$ Torrent crashes while it is minimized and the user has been idle at the computer for more than 1 minute, this option will cause  $\mu$ Torrent to automatically restart, and a notification of the crash (without a crash dump or any personally identifiable information) will be sent to the developers. If  $\mu$ Torrent crashes more than once within an hour, this option will not cause it to automatically restart again after the first crash, as such frequency

of crashes is indicative of some important underlying problem that should not be ignored by the user. In this situation,  $\mu$ Torrent will fall back to the regular behavior (as if this option were disabled), where it displays a crash dialog that allows the user to choose how to proceed.

- **gui.color\_progress\_bars:** This option enables or disables color in the torrent progress bars.
- **gui.combine\_listview\_status\_done:** This option will show you the progress of the download (% done downloading) inside the colored status bar/column.
- **gui.compat\_diropen:** If you experience abnormal behaviors while browsing directories in  $\mu$ Torrent, such as a blank browsing dialog, try enabling this option.
- **gui.default\_del\_action:** This option tells  $\mu$ Torrent how it should remove torrent jobs when pressing the **Remove button** or **Delete** on your keyboard. Note that any value above 3 will cause the "Remove" button and the **Delete** button on your keyboard to do nothing in  $\mu$ Torrent. To be safer, you'd best be setting this option in the GUI through **the toolbar method**.
  - **0** means "Remove"
  - **1** means "Remove and delete .torrent"
  - **2** means "Remove and delete Data"
  - **3** means "Remove and delete .torrent + Data"
- **gui.delete\_to\_trash:** Enabling this option tells  $\mu$ Torrent to attempt to delete files to the Recycle Bin rather than directly erasing them from the disk. It is easier to set this option in the GUI through **the toolbar method**.
- **gui.enable\_comments:** This option enables or disables the torrent comment feature.
- **gui.graph\_legend:** This option tells  $\mu$ Torrent to draw a legend over the graphs displayed in the **Speed tab** to describe each of the lines

drawn on the graph.

- **gui.graph\_overhead**: If enabled, this option tells  $\mu$ Torrent to draw communication overhead lines in the **Speed** tab's transfer rate graphs. Otherwise, only the "*Network Overhead*" graph will display information about communication overhead.
- **gui.graph\_tcp\_rate\_control**: This option tells  $\mu$ Torrent to draw the TCP rate control/limit graph on the download speed graphs
- **gui.graphic\_progress**: This option tells  $\mu$ Torrent to draw a progress bar for each torrent job in the torrent jobs list, behind **the Done column**.
- **gui.log\_date**: This option causes the date to be included in the timestamp shown in the **Logger** tab.
- **gui.overhead\_in\_statusbar**: This option shows the protocol-network overhead in the status bar.
- **gui.piecebar\_progress**: If enabled, this option tells  $\mu$ Torrent to draw the **lower Downloaded bar** as the progress bar for each torrent job in the torrent jobs list, behind **the Done column**. This option works only if **gui.graphic\_progress** is enabled, and will hide the percentage from the column.
- **gui.report\_problems**: If enabled, this option tells  $\mu$ Torrent to report hangs in the user interface thread back to the  $\mu$ Torrent servers anonymously. The information sent is not personally identifiable, but can assist the developers in fixing (or identifying the cause of) the user interface hang. Whenever a report is sent to the server, a message is added to the **Logger** tab.
- **gui.show\_av\_icon**: This option shows an antivirus icon when Plus version is installed.
- **gui.show\_devices**: This option show devices pane in sidebar.
- **gui.show\_notorrents\_node**: This option will show a page in the

sidebar when no torrents are loaded.

- **gui.show\_player\_node:** This option puts a new item in the sidebar when streaming a video in the client.
- **gui.show\_plus\_upsell:** This option shows the "Plus" box on the sidebar.
- **gui.show\_rss\_favicons:** This option shows favicons for your RSS feeds
- **gui.show\_status\_icon\_on\_dl\_list:** This option shows status icons in the main listview of torrents.
- **gui.tall\_category\_list:** This option toggles the **Category List**'s height between short and tall. When taller, the Category List displaces the **Detailed Info Pane**'s left-hand side. When shorter, the Category List's lower section is displaced by the Detailed Info pane. A taller list might be more optimal for users with many labels and RSS feeds
- **gui.transparent\_graph\_legend:** If enabled, this option tells  $\mu$ Torrent to draw a transparent background behind the legend (otherwise, the background is opaque).
- **gui.update\_rate:** This option controls the amount of time between each update of the  $\mu$ Torrent main window. The higher it is, the less frequently  $\mu$ Torrent updates the main window, meaning that if you select 1000, the information displayed on the main window is at most 1000 milliseconds (1 second) old. For users of slower computers, you might want to increase this number to decrease resource usage when the main window is displayed. Any value below 500 will be ignored (and 500 will be used instead).
- **gui.use\_fuzzy\_dates:** This option shows inexact dates instead of timestamps (e.g. "5 minutes ago")
- **ipfilter.enable:** This option, when enabled, tells  $\mu$ Torrent to load

`ipfilter.dat` and apply the rules on connections established after it is loaded. Note that disabling and re-enabling this option will force  $\mu$ Torrent to reload `ipfilter.dat`.

- **isp.bep22:** This option enables Local Tracker Discovery according to `BEP22`, allowing  $\mu$ Torrent to attempt to discover ISP-local trackers via a series of reverse DNS lookups. The ISP-local tracker can return a list of peers and caches (most likely ISP-local). Note that if your ISP is known to interfere with BitTorrent traffic, careful consideration should be taken in deciding to enable this option. Announcing to a ISP-hosted tracker indicates to the ISP that you are using BitTorrent, and as such, can make it easier for the ISP to interfere. Private torrent jobs are not announced to local trackers.
- **isp.fqdn:** If your ISP does not return a correct reverse-DNS name, this allows you to set your reverse lookup name for the purposes of `BEP22`
- **isp.peer\_policy\_enable:** This option enables peer policy functionality, which sets weights to different IP ranges.
- **isp.peer\_policy\_override:** This option overrides the peer policy.
- **isp.peer\_policy\_url:** This option sets a URL to the ISP's peer policy.
- **isp.primary\_dns:** This option sets the primary DNS server IP of your ISP.
- **isp.secondary\_dns:** This option sets the secondary DNS server IP of your ISP.
- **left\_rail\_offer\_enabled:** This option enables new "Offers" (Ads) by BitTorrent Inc located at the left pane. Disable it if you like not to be alerted to new offers.
- **logger.log\_upnp\_to\_file:** This option logs debug output of UPnP to a file (warning: spammy).

- **net.bind\_ip**: If your computer setup requires that you use a specific LAN adapter for incoming connections, you may specify that adapter's IP address here.
- **net.calc\_rss\_overhead**: This option applies ratelimits to RSS traffic as well.
- **net.calc\_tracker\_overhead**: This option applies ratelimits to tracker traffic (warning: could break tracker communication under load).
- **net.discoverable**: If enabled, this option tells  $\mu$ Torrent to listen on one of a sequence of well-known ports for incoming connections in addition to the **standard** and **alternative** listening ports. Because the sequence of ports is well-known to applications attempting to interface with  $\mu$ Torrent, it allows for such applications to connect to  $\mu$ Torrent with less effort on the user's part.
- **net.disable\_incoming\_ipv6**: This option blocks all incoming IPV6 connections.
- **net.friendly\_name**: Local network UPnP device name.
- **net.limit\_excludeslocal**: This option decides whether  $\mu$ Torrent should apply the **Transfer Cap** limits to traffic between itself and peers on the local network. Peers are considered local if they are discovered by Local Peer Discovery, or if they are on the same LAN as the client.
- **net.low\_cpu**: Enabling this option reduces CPU usage slightly. You may achieve faster speeds with this option disabled. In general, this option is useless for most people unless they have extremely fast connections.
- **net.max\_halfopen**: This option specifies how many connections  $\mu$ Torrent should attempt to establish simultaneously at any given time. On systems running Windows XP with Service Pack 2 (SP2) or newer, if your TCPIP.sys file is unpatched, you should leave this option at its default value.

- **net.outgoing\_ip**: If your computer setup requires that you use a specific LAN adapter for outgoing connections, you may specify that adapter's IP address here. Note that Windows will sometimes ignore this setting and use other adapters due to their binding orders in Windows. To fix this, read Microsoft's knowledge base article [KB894564](#).
- **net.outgoing\_max\_port**: This sets the upper limit for the outgoing port range. If this option is set to some invalid port number or some value less than [net.outgoing\\_port](#), it gets ignored, and only [net.outgoing\\_port](#) gets looked at (meaning the outgoing port "range" will actually be a single outgoing port).
- **net.outgoing\_port**: This option specifies the port that  $\mu$ Torrent should use to make outgoing connections. Normally,  $\mu$ Torrent selects a port from the ephemeral port range at random. "This can be used with full cone NAT routers to reduce the number of NAT table entries and thus prevent caches on some router models. When the outgoing port is bound to the same as the incoming port that might even solve NAT problems on full cone NAT routers" ([Advanced Network Settings](#) on AzureusWiki). This option only works on Windows 2000 and above. This option is ignored if it is not a valid port number.
- **net.upnp\_tcp\_only**: This option disables automatic forwarding of the listening port for UDP via UPnP, telling  $\mu$ Torrent to forward the port for TCP only. This fixes an issue with some broken routers that overwrite the TCP forwarding with the UDP forwarding.
- **net.utp\_dynamic\_packet\_size**: If enabled, this option allows  $\mu$ Torrent to adjust the uTP packet size in response to connection conditions detected through information gathered by uTP, changing up to as often as [net.utp\\_packet\\_size\\_interval](#) allows. If disabled,  $\mu$ Torrent uses the initial packet size for all uTP communication, as set by [net.utp\\_initial\\_packet\\_size](#).
- **net.utp\_initial\_packet\_size**: This controls the initial size of the uTP packets that  $\mu$ Torrent uses when initiating a uTP connection. If

`net.utp_dynamic_packet_size` is enabled, packet sizes can change dynamically during the lifetime of the uTP connection, depending on the connection conditions; this option only controls how  $\mu$ Torrent starts off. This option is interpreted as a multiplier of 150 bytes, so please enter it as such. Any value below 1 will be ignored (and 1 will be used instead), and any value above 8 will be ignored (and 8 will be used instead). Effectively, that means that the initial packet sizes selectable by the user are the multiples of 150 bytes between (and including) 150 bytes and 1200 bytes.

- **`net.utp_packet_size_interval`**: This controls how often uTP alters its packet size in response to network conditions, assuming `net.utp_dynamic_packet_size` is enabled. This value is interpreted in seconds, so please enter it as such.
- **`net.utp_receive_target_delay`**: This controls the threshold detected connection receive delay that, if surpassed, will cause  $\mu$ Torrent to throttle back on bandwidth usage. The higher this option is set, the more forgiving  $\mu$ Torrent will be toward connection delays, meaning that it will be less likely to throttle back on bandwidth usage. Receive delay is detected by tracking the changes in the deltas between uTP packet timestamps and packet receive times. This option is interpreted in milliseconds, so please enter it as such.
- **`net.utp_target_delay`**: This option controls the threshold detected connection send delay that, if surpassed, will cause  $\mu$ Torrent to throttle back on bandwidth usage. The higher this option is set, the more forgiving  $\mu$ Torrent will be toward connection delays, meaning that it will be less likely to throttle back on bandwidth usage. Send delay is the receive delay as observed by recipient uTP peers, which is reported back to the client by the recipient peers. This option is interpreted in milliseconds, so please enter it as such.
- **`offers.content_offer_autoexec`**: When enabled, if torrent offer (featured content) has autoexec flag AND is signed correctly by us,  $\mu$ Torrent will auto-execute the content.

- **offers.content\_offer\_url**: Used to test install-time content offers
- **peer.disconnect\_inactive**: Enabling this option tells µTorrent to disconnect from a peer that is not transferring with you after **peer.disconnect\_inactive\_interval** seconds of inactivity. A peer gets disconnected by this option only if **the connection limit** has been reached.
- **peer.disconnect\_inactive\_interval**: This option sets the amount of time µTorrent should wait before breaking an inactive connection. This value is interpreted in seconds, so please enter it as such. Any value below 300 will be ignored (and 300 will be used instead).
- **peer.lazy\_bitfield**: Some ISPs block seeding by looking for the complete bitfield and closing the connection. When enabled, µTorrent does not send the complete bitfield, but a sample of it, so as to prevent blocking of seeding.
- **peer.resolve\_country**: Enabling this option tells µTorrent to use an Internet database of IP addresses (a DNSBL) to determine a peer's country. Even if the settings directory contains **flags.conf** and **flags.bmp**, this option will take precedence, and the internal flag images will be used instead.
- **queue.dont\_count\_slow\_dl**: Enabling this option tells µTorrent to ignore slow downloading torrent jobs as part of the queue. If a torrent job is downloading at less than the value specified by **queue.slow\_dl\_threshold**, it will not prevent the next item in the queue from starting.
- **queue.dont\_count\_slow\_ul**: Enabling this option tells µTorrent to ignore slow uploading torrent jobs as part of the queue. If a torrent job is uploading at less than the value specified by **queue.slow\_ul\_threshold**, it will not prevent the next item in the queue from starting.
- **queue.prio\_no\_seeds**: Enabling this option gives torrent jobs without seeds higher priority when seeding than other torrent jobs.

- **queue.slow\_dl\_threshold:** The rate below which  $\mu$ Torrent should consider a torrent job to be downloading slowly. If  $\mu$ Torrent is downloading at a rate above this value, it is considered to be actively downloading. This value is interpreted in bytes per second, so please enter it as such.
- **queue.slow\_ul\_threshold:** The rate below which  $\mu$ Torrent should consider a torrent job to be uploading slowly. If  $\mu$ Torrent is uploading at a rate above this value, it is considered to be actively uploading. This value is interpreted in bytes per second, so please enter it as such.
- **queue.use\_seed\_peer\_ratio:** When this option is enabled,  $\mu$ Torrent will determine the seeding queue order based on the ratio of the number of seeds to the number of peers connected in the swarm. The lower the seed:peer ratio is for a torrent job, the higher priority it will be given in the seeding queue. If a torrent job has 0 peers and `queue.dont_count_slow_ul` is disabled, it will be given the lowest priority. Otherwise, if the aforementioned option is enabled, the torrent job is treated as if there is 1 peer in the swarm.
- **remove\_torrent\_files\_with\_private\_data:** This option deletes torrents in the .torrent files directory that are not loaded into client.
- **rss.feed\_as\_default\_label:** When this option is enabled,  $\mu$ Torrent will use an RSS feed's name as the default label for any torrent jobs added without a label from the RSS feed.
- **rss.smart\_repack\_filter:** This option tells  $\mu$ Torrent to select an RSS item designated as REPACK over an item without the REPACK designation if both show up in the RSS feed.
- **rss.update\_interval:** This option sets the length of time  $\mu$ Torrent should wait between each RSS feed update check. This value is interpreted in minutes, so please enter it as such. Any value below 5 will be ignored (and 5 will be used instead).
- **sponsored\_torrent\_offer\_enabled:** This options enables new

"Offer"-torrents by Bittorrent Inc located at the top of your main view. Only new offers will be disabled, and you can skip/cancel them on screen at will. Disable it if you like not to be offered new torrents.

- **store\_torr\_infohash:** This option saves the .torrent file as INFOHASH.torrent (e.g. ABCDEF1234567890ABCD.torrent).
- **streaming.failover\_set\_percentage:** This option sets the maximum size of the failover set, expressed as percentage of the total number of peers.
- **streaming.min\_buffer\_piece:** This option sets minimum number of pieces to hold in the streaming buffer.
- **streaming.safety\_factor:** This option ensures download rate is faster in % than the calculated rate needed.
- **sys.enable\_wine\_hacks:** This option enables several workarounds for bugs found in Wine (like list-view flickering, or improper display of files list-view in Add New Torrent dialog). This option has no effect on Windows. For changes to this option to take effect, you must restart µTorrent. This applies only if we're on Wine.
- **webui.allow\_pairing:** This option allows device/service to pair with the client and control it via WebUI.
- **webui.token\_auth:** This option enables the token authentication system for the Web UI, which is a method for preventing cross-site request forgery attacks that use the authenticated browser session to issue commands to µTorrent. This option breaks backwards compatibility with applications that are unaware of the token system.
- **webui.token\_auth\_filter:** If `token_auth = true`, then this **token\_auth\_filter** changes the scope of token authentication: 0 means that it applies to all connections, 1 means that it applies only to remote connections (i.e. not localhost).



[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >  
[Preferences](#) > [Advanced](#)

# UI Extras

## Speed Popup List

- **Override automatic speed popup list** allows you to define the lists of speeds that are displayed when `right-click`ing the corresponding item in the status bar on the main window. If left unchecked,  $\mu$ Torrent automatically generates the list in context, based on the current limit.
  - **Upload speed list** defines the upload speed list. -1 adds a separator to the list. 0 means **Unlimited**. Any other positive value is interpreted normally (as that value). Each value must be separated with a comma. The values are interpreted in KiB/s, so please enter them as such.
  - **Download speed list** defines the download speed list. -1 adds a separator to the list. 0 means **Unlimited**. Any other positive value is interpreted normally (as that value). Each value must be separated with a comma. The values are interpreted in KiB/s, so please enter them as such.

## Persistent Labels

- The field here allows you to specify permanent labels that remain regardless of the number of torrent jobs there are actually using that label. Multiple persistent labels may be entered, but must be separated with a | (vertical pipe). Note that persistent labels are given higher priority than normal labels in the `category list`, and so, are displayed above their normal counterparts.

## Search Engines

- This edit control allows you to edit the list of search engines that  $\mu$ Torrent can use when using the built-in torrent search facility. One search engine may be entered per line, and must be entered as follows (where ?search= is arbitrary and depends on the search engine being used):  
Name | http://domain/path?search=

- If found in the URL,  $\mu$ Torrent will replace the following tokens as described:
  - %s is replaced with the entered search query. If this is not found in the search URL, then the query will be appended to the end of the URL instead.
  - %v is replaced with the client user agent ID.
- In order to add custom search engines to the list on your own, you should have some basic understanding of HTML and how HTTP GET works. The topics themselves are outside the scope of this manual. The alternative would be to check whether your favorite search engine serves .btsearch files that  $\mu$ Torrent can use to automatically add the search engine to the list.
- Blank lines are displayed as separators in the search engine menu.
- If you leave the entire field blank, the [search widgets in the toolbar](#) will be hidden from view.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >  
[Preferences](#) > [Advanced](#)

# Disk Cache

The disk cache is used to keep frequently accessed data in memory to reduce the number of reads and writes to the hard drive.  $\mu$ Torrent normally manages the cache automatically, but you may change its behavior by modifying these settings. Because of the nature of the settings, caution should be exercised when modifying the settings, as bad settings may be detrimental to performance. Disk cache behavior can be viewed in the [Speed tab](#).

## Disk Cache

- **Override automatic cache size and specify the size manually** allows you to set a new maximum memory usage for the disk cache. Normally,  $\mu$ Torrent automatically adjusts the cache size based on your current download speed. This value is interpreted in MiB, so please enter it as such. Any value below 1 will be ignored, and 1 will be used instead.
- **Reduce memory usage when cache is not needed** flushes unused parts of the disk cache when  $\mu$ Torrent is not transferring data.

## Advanced Cache Settings

- **Enable caching of disk writes** allows  $\mu$ Torrent to use the disk cache to store data in memory before writing it to disk. This option has the effect of decreasing the frequency of writes to disk.
  - **Write out untouched blocks every 2 minutes** makes  $\mu$ Torrent write to disk the 16 KiB blocks of data in memory that haven't been touched for 2 minutes or longer and don't fill up a piece. This option has the effect of decreasing  $\mu$ Torrent's memory usage while increasing the frequency of writes to disk.
  - **Write out finished pieces immediately** makes  $\mu$ Torrent write a piece to disk once it is completed in memory. This option has the effect of decreasing  $\mu$ Torrent's memory usage while increasing the frequency of writes to disk.

- **Enable caching of disk reads** allows  $\mu$ Torrent to use the disk cache to store frequently read data into memory from disk. This option has the effect of decreasing the frequency of reads from disk.
  - **Turn off caching if the upload speed is slow** disables the read caching if uploading is below 40 KiB/s. This option has the effect of decreasing  $\mu$ Torrent's memory usage while increasing the frequency of reads from disk.
  - **Remove old blocks from the cache** removes chunks of read cache data from memory if they haven't been accessed for 10 minutes in order to make room for other data. This option has the effect of decreasing  $\mu$ Torrent's memory usage while increasing the frequency of reads from disk.
  - **Increase automatic cache size when cache thrashing** increases the disk cache size when reading from disk increases (most often due to increased upload speeds). If you override the automatic cache size, this option is not needed.

[« Previous](#)   [Next »](#)

[µTorrent User Manual > Appendix A: The µTorrent Interface >](#)

[Preferences](#)

# Web UI

The Web UI is used to allow external applications to control  $\mu$ Torrent remotely. The default Web UI for  $\mu$ Torrent is accessible to modern browsers that support AJAX.

## Web UI

**Enable Web UI** will enable the web interface if checked. When enabled,  $\mu$ Torrent should automatically download the required [webui.zip](#) file from the  $\mu$ Torrent server. If it doesn't, you can still download it manually from [the webpage](#) and properly place it in [the  \$\mu\$ Torrent settings directory](#).

A detailed guide for using the WebUI is [available here](#)

## Authentication

- The **Username** field allows you to specify a username to log into the web interface, and is required in order to use the web interface.
- The **Password** field sets the password for this username. It may be left blank if you do not wish to password-protect the web interface.
- **Enable Guest account with username** enables or disables the guest account, a user that can view (but not edit) the torrent jobs currently in your torrent jobs list. If enabled, a username for the guest account must be specified in order for it to work.

## Connectivity

- **Alternative listening port**, if enabled, tells  $\mu$ Torrent to use the specified port to listen to Web UI requests (and the [internal tracker](#) if enabled). This option is especially useful for those who randomize their ports each time  $\mu$ Torrent starts. If left disabled,  $\mu$ Torrent will

serve the web interface over the same port as its normal listening port. You must forward this port manually, even if you use UPnP to forward the normal  $\mu$ Torrent listening port.

- **Allow access only from these IPs** allows you to limit the IPs that are allowed to connect to the web interface. Multiple IPs may be specified, but they must be separated by commas. IP ranges may be specified using CIDR notation.

## Download Directories

- **Add/Remove** You may add or remove any directory you wish to enable remote access to.

[« Previous](#)   [Next »](#)

[µTorrent User Manual](#) > [Appendix A: The µTorrent Interface](#) >  
[Preferences](#) > [Advanced](#)

# Run Program

The "Run Program" dialogue allows you specify a program or a batch file that you want uTorrent to execute when any torrents download completes, or on any number of events while you download or seed. Here is an example:

***"d:\program files\utorrent\run-program.cmd" "%D"***

***This will cause utorrent to run the "run-program.cmd" batch file wit the file-path as a parameter to it.***

A full list of the possible events and parameters are available on this dialog.

## Run Program

- **Run this program when a torrent finishes** edit in the command line you want uTorrent to execute when download completes.
- **Run this program when a torrent changes state** edit in the command line you want uTorrent to execute when there is any change in the torrent state.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface](#)

# RSS Downloader

RSS web feeds deliver information and content in such a way that allows one to track updates using an aggregator.  $\mu$ Torrent can aggregate and parse an RSS feed, allowing one to automatically download newly released .torrent files that match a specified filter. Such feeds must give direct links to the .torrent files in at least one of either the <guid>, <link>, or <enclosure> tags in order to be used correctly. To access the RSS Downloader, select "*Options*" then "*RSS...*" in the **main menu**, or press **Ctrl + R** on the keyboard.

A tutorial is available in Chapter 02 - "**Downloading Using the RSS Downloader**"

- **Favorites**
- **History**

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[RSS Downloader](#)

# Favorites

The **Favorites** tab manages filters that allow  $\mu$ Torrent to automatically download specific torrents from a RSS feed.

This dialog is also explained in the example in Chapter 02 - "Downloading Using the RSS Downloader".

- The listbox displays all filters that  $\mu$ Torrent will use when parsing the RSS feeds. If the checkbox next to a filter is checked, it will be used by  $\mu$ Torrent to pick out a torrent to download. Otherwise, the filter is ignored by  $\mu$ Torrent. Filters can be sorted using **drag-and-drop**.
- The **Add** button allows you to enter a new filter for  $\mu$ Torrent to use when parsing RSS feeds. After you press "Add", you must name the filter immediately. You can edit this name later by selecting the filter from the listbox and pressing **F2** on your keyboard. An alternative is to **double-click** the filter and rename it directly.
- The **Delete** button allows you to delete the filter that is currently selected.
- The **?** (question mark) button tests displays the last time the selected filter matched a torrent. It also displays the last 4 torrents matched (if any) and any currently-matching torrents from all applicable feeds. This can be used to double-check your filter.
- The **Filter** field allows you to tell  $\mu$ Torrent what you want to match. **Wildcards** can be used to generalize the filter. Note that you cannot specify the season or episode number in this filter if it is decoded by  $\mu$ Torrent. You must use the episode number box, or enable "*Filter matches original name instead of decoded name*". This field is case insensitive.
- The **Not** field allows you to specify text that, if matched by a torrent, will cause  $\mu$ Torrent to ignore it. **Wildcards** can be used to generalize

the filter. Note that you cannot specify the season or episode number in this filter if it is decoded by  $\mu$ Torrent. You must use the episode number box, or enable "*Filter matches original name instead of decoded name*". This field is case insensitive.

- The **Save in** field allows you to specify where you want  $\mu$ Torrent to download the torrent contents to. If this field is specified,  $\mu$ Torrent will automatically start the torrent job. Note that if you already have a default download location set, you do not have to set a directory here.
- The **Feed** dropdown menu allows you to select a particular feed that you want the filter to apply to. If you want  $\mu$ Torrent to apply this filter to every feed, leave this dropdown menu as (All).
- The **Quality** menu allows you to select the quality of the torrents, if it is a video. If what you want to filter is not a video, leave this option as ALL. Note that setting this option is not an absolute necessity, even if you are trying to download a video.
- The **Episode Number** field allows you to specify specific episodes to download. This field supports ranges, so if you want to download episodes 1 through 26 of season 3 of some series, you input 3x1-26 into this field. If you want to specify the ending season in addition to the starting season, you can enter something like 3x1-4x15 into the field. If you want to specify the beginning of a range, but not the end (you would like  $\mu$ Torrent to continue every subsequent episode, including in later seasons), something like 3x1- would suffice. Notice the - (hyphen) after the 1. If you prefer, S3E1 can also be used to specify episodes instead of the above-shown 3x1.
- **Filter matches original name instead of decoded name** forces  $\mu$ Torrent to apply the filter on the torrent name itself. Normally,  $\mu$ Torrent parses the torrent name and applies the filter on the decoded name, which excludes certain things like quality and episode number.

- **Don't start downloads automatically** tells  $\mu$ Torrent to add torrent jobs matching the favorite in stopped mode instead of starting them automatically.
- **Give download highest priority** places a matching torrent at the top of the download queue (#1). Because this is not a forced start, if you pass your maximum number of active torrent jobs after this torrent job gets started, the started torrent closest to the end of the queue will be stopped and will have to wait on the queue.
- **Smart ep. filter** causes  $\mu$ Torrent to download the first version of each new episode that matches your filter. Note that you should not enable this option if the episode number is not parseable (when the **Episode** column is empty).
- The **Minimum interval** dropdown menu sets the minimum interval between each time the filter is matched. For example, if set to 2 days,  $\mu$ Torrent will not attempt to match the filter again until at least 2 days have passed.
  - **(match always)** tells  $\mu$ Torrent to check the filter for matches every time it checks the RSS feeds.
  - **(match only once)** tells  $\mu$ Torrent to use the filter only once. After a torrent job is added to the torrent jobs list by the filter,  $\mu$ Torrent won't check the filter for any more matches.
- The **Reset** button forces  $\mu$ Torrent to forget when it last matched the filter.
- The **Label for new torrents** combobox tells  $\mu$ Torrent to set any torrent that matches this filter to the specified label.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface >](#)

[RSS Downloader](#)

# History

The **History** tab displays the last 500 torrents matched. Entries can be manually deleted by use of the **Delete** button at the bottom, or by pressing **Delete** on the keyboard. Note that **left-click**ing on a column will sort the items in the list by that column, alternating between ascending and descending sort for each additional click. You can select what columns you wish to see by **right-click**ing the column headers. If you wish to reset all changes you've made to the columns, you can select "*Reset*" in that context menu. The following is a description of each column:

- **Download Date** is the date that the torrent job was matched.
- **Feed** is the feed from which the added torrent job was matched.
- **Full Name** is the unparsed name of the torrent (before any information is extracted or decoded).
- **URL** displays the URL that the source .torrent file is located at.

## Context Menu

- **Delete** will delete the selected entries from the RSS history.
- **Open URL in Browser** will open the linked URL(s) in your default web browser.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface](#)

## Miscellaneous

There are several other elements in the  $\mu$ Torrent interface that should be explained, but do not fit into the previous sections of the manual.

- [Add New Torrent](#)
- [Add/Edit RSS Feed](#)
- [Create New Torrent](#)
- [Torrent Properties](#)
  - [General](#)
  - [Advanced](#)
- [Tray Icon](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix A: The μTorrent Interface >](#)

Miscellaneous

# Add New Torrent

The Add New Torrent dialog allows you to change several options for the torrent job before it is added to your torrent jobs list. This dialog is displayed only if **the relevant option** is enabled in the Preferences. Files selected/deselected in this dialog can be deselected/reselected in the **Files tab's** context menu.

When adding **magnets**, you can choose to wait while  $\mu$ Torrent retrieves the torrent information. At this time, a revolving "waiting" icon will be display at the bottom (a temporary message "**waiting for torrent information**" might be displayed as well in future versions...)

## Save In

- The combobox allows you to specify the location you wish to download the torrent contents. Previously used locations are saved in the combobox and can be used again.

## Create Subfolder

- If checked a containing folder will be created in your download directory and the torrent's contents will be downloaded into it. If unchecked, the torrent's contents will be placed directly into your download folder in their original file hierarchy.

## Name

- The combobox allows you to change the name of the torrent. It will be used as the file name in case of a single file torrent. For torrents with multiple files - this will be the name of the subfolder under which the files will be saved.

## Torrent Options

- The **Skip hash check** checkbox tells µTorrent to not bother checking existing data for corruption. It is strongly recommended that you do not use this option, as corruption can sometimes happen unknowingly, outside of one's own control. Sharing corrupt pieces is detrimental to the swarm.
- The **Start torrent** checkbox tells µTorrent to start the torrent job after it is added to the torrent jobs list. If unchecked, the torrent job will be added in Stopped mode.
- The **Label** combobox allows you to set a label for the torrent job being added. Existing labels are displayed in the combobox, but you are free to create a new label if you wish.
- The **Add to top of queue** checkbox will give the torrent job a queue number of 1. If combined with the **Start torrent** option, this will start the torrent job immediately.

## Torrent Contents

- **Name** displays the name of the torrent job, taken from the .torrent file.
- **Comment** shows the comment placed on the .torrent file job by its creator.
- **Size** displays the total amount of space required for the files being downloaded. Selective file downloading is taken into account, so only the data you select to be downloaded will be counted. When files are skipped, the total size of the torrent contents (not taking selective file downloading into account) is displayed next to the selected size. The last number in the parentheses displays the amount of space available in the disk that the torrent contents are being downloaded to.
- **Date** shows the day and time at which the .torrent file was created.
- The **Select All** button will mark all files to be downloaded.

- The **Select None** button will mark all files to be skipped.
- The **Files** list-view is a list of all files in the .torrent file, and allows you to select which files you want to download. The following is a description of each column:
  - **Name** column displays the filename. A checkmark to the left of the filename means the file will be downloaded, while an empty checkbox means the file will not be downloaded. Pressing **Space** on the keyboard toggles this for the selected file(s).
  - **Path** is the location that the file will be downloaded to on disk. If this column is blank, then the file will be downloaded into the torrent job's **Save As** location. If this column displays a partial path (not a full path that includes a drive root), then that is the subdirectory in the **Save As** directory that the file will be saved to.
  - **Size** column displays the size of the file.

**right-click**ing on the list will display the following items in a context menu:

- **Select** will mark the selected file(s) to be downloaded.
- **Deselect** will mark the selected file(s) to be skipped.

---

- **Select All** will mark all files to be downloaded.
- **Deselect All** will mark all files to be skipped.

The **Advanced...** button brings up the **Torrent Properties** dialog.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Miscellaneous](#)

## Add/Edit RSS Feed

The Add/Edit RSS Feed dialog allows you to enter a new feed for  $\mu$ Torrent to monitor, or edit an existing feed. Feeds must contain direct links to .torrent files in at least one of either the <guid>, <link>, or <enclosure> tags in order to be used correctly.

### Feed

- The **Feed URL** field specifies the URL that the RSS feed is located at. Depending on the method used by the website for authenticating users (if any), the URL may have to be entered **in a specific format**.
- The **Custom Alias** field allows you to give a custom name to the feed. If unchecked,  $\mu$ Torrent will attempt to automatically detect the feed name based on the title specified in the feed itself.

### Subscription

- **Do not automatically download items** tells  $\mu$ Torrent not to automatically create a simple favorite for the RSS feed to download any torrent in the feed.
- **Automatically download items published in feed** tells  $\mu$ Torrent to create a simple favorite for the RSS feed that matches and downloads any item in the feed (uses \* as the filter).
  - **Use smart episode filter** applies the smart episode filter to the simple favorite.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Miscellaneous](#)

# Create New Torrent

So you've finished setting everything up, downloaded some torrents, and are now wondering how you can share your own files with other people?  $\mu$ Torrent makes the process very simple, and after only a few short steps you will be able to share your files easily. Despite its outward appearance, the .torrent file creation facility's simplicity does not mean it is lacking in functionality, and in most cases, it is all people need in a .torrent file creation utility. The only required step in creating a .torrent file is to select a valid source, though you should configure the rest of the settings appropriately if needed.

After you create the .torrent file, you need to find a way to share it with whoever you want. If you're only targeting a few people, it might be easier to send the file to them directly. If you're targeting the general public, you should upload it to a website that allows .torrent file uploads. Public .torrent index sites often allow for this. The .torrent file can also be uploaded to private trackers, but you should be aware that some private trackers require that you reupload the .torrent file from their tracker in order for it to work.

## Select Source

- The combobox displays the path of the file or directory that will be made into a .torrent file. You can edit this path manually if you know the file or directory's location. Previously used locations are saved in the combobox and can be used again.
- The **Add file** button lets you select a specific file and automatically places its path in the combobox above. Selecting a single file disables the "*Skip Files*" feature.
- The **Add directory** button lets you select a specific directory and automatically places its path in the combobox above. Only by using this button can you make use of the "*Skip Files*" feature.

- The **Skip Files** field allows you to skip files using specific filters. This field only gets activated if you use the "*Add directory*" button. **Wildcards** can be used to specify multiple files.

## Torrent Properties

- The **Trackers** edit control lists all of the trackers the .torrent file will use. Only HTTP, HTTPS, and UDP trackers are supported. Trackers are separated by a newline. Those that are not separated by blank lines will be used as backup trackers for adjacently listed trackers. Backup trackers will not be used unless the previously listed tracker fails. Note that having µTorrent use multiple trackers simultaneously does not necessarily improve the .torrent itself, so don't be discouraged from selecting some trackers to be backup trackers. If you're not sure what tracker to use, try checking your favorite .torrent tracker for the tracker URL to use when creating new .torrents for it. Alternatively, you can use the **embedded tracker** in µTorrent here. This edit control can be left blank if you wish to use only DHT to track the .torrent file, just be sure that you do not make the .torrent file private.
- The **Web Seeds** edit control lists all the web seeds the .torrent file will use. Only HTTP, HTTPS, and FTP web seeds are supported. Web seeds are separated by a newline. For single-file torrent contents, the target file can be linked to directly (even if the file name differs from the name described in the .torrent file). This isn't necessary, though; for both single and multiple file torrent contents, the directory containing the desired files can be listed, and clients that support web seeds should automatically append the path and filenames described in the .torrent file onto the listed directory to obtain the full URL to the desired files.
- The **Comment** field allows you to place whatever comment you want in the .torrent file.
- The **Piece Size** dropdown menu allows you to select the size of each piece in the .torrent file. Selecting too small a piece size for

large source files creates a needlessly large .torrent file and extreme protocol communication overhead, while using too large a piece size means more wasted bandwidth for people who often experience hashfails, since they'd have to redownload entire pieces over for each hashfail that occurs. Unless you are absolutely sure you understand what the option is used for, and the consequences that come with using it, it is recommended that you let  $\mu$ Torrent automatically detect the piece size for you.

## Other

- **Start seeding** tells  $\mu$ Torrent to automatically add the torrent job to the torrent jobs list and start it. Note that if you plan on uploading the .torrent file to a private tracker, you might have to redownload the file from their tracker before it works, so this option might be useless to you if that is the case.
- **Preserve file order** tells  $\mu$ Torrent to create the .torrent file with files and directories sorted by name, in ascending order. By default,  $\mu$ Torrent sorts files in the .torrent file by size, in descending order.
- **Private torrent** tells  $\mu$ Torrent to set the 'private' flag in the .torrent file, which tells clients that support this flag to not announce this torrent job over DHT, and disables LPD and PEX as well. Accordingly, if you plan on leaving the **Trackers** edit control blank in order to make the .torrent file use DHT, you should not enable this option.
- **Create Encrypted** tells  $\mu$ Torrent to encrypt the .torrent file.

## Related

- **Website** allows you to manually enter website info to be stored as metadata within the .torrent file.
- **Rss** allows you to manually enter RSS info to be stored as metadata within the .torrent file.

- **Similar Torrents** allows you to manually enter similar torrent info to be stored as metadata within the .torrent file.

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Miscellaneous](#)

# Torrent Properties

In some cases, you might want a torrent job to behave differently from the default settings, maybe in terms of speed, or maybe how it obtains its peers. With the Torrent Properties dialog, you can edit some of these settings.

- [General](#)
- [Advanced](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Miscellaneous](#) > [Torrent Properties](#)

# General

## Trackers

- The **edit control lists all the trackers the torrent job(s) will use, and is interpreted identically to how the "Create New Torrent" dialog's "Trackers" edit control** is interpreted. If a private torrent job's trackers list is modified, all existing peer connections for that torrent job will be dropped.

## Bandwidth Settings

- The **Maximum upload rate** field will limit the maximum rate at which  $\mu$ Torrent will upload for the selected torrent job(s). Setting the value to 0 is equivalent to setting it to unlimited. Setting the value to anywhere between 1 and 3 will limit the download speed for the torrent job to 12 times the set upload speed, and you will see a "[Limited]" message in the **Transfer tab**. This value is interpreted in KiB/s, so please enter it as such. This limit will not override the **global maximum upload rate**.
- The **Maximum download rate** field will limit the maximum rate at which  $\mu$ Torrent will download for the selected torrent job(s). Setting the value to 0 is equivalent to setting it to unlimited. This value is interpreted in KiB/s, so please enter it as such. This limit will not override the **global maximum download rate**.
- The **Number of upload slots** field defines the maximum number of peers that  $\mu$ Torrent will upload to at any given moment for the selected torrent job(s). This value will override the **default value**. Leaving this blank tells  $\mu$ Torrent to use the default settings.
- The **Minimum number of available seeds** field defines the minimum number of seeds This value will override the **default value**. Leaving this blank tells  $\mu$ Torrent to use the default settings.

## Seeding Goal

- **Override default settings** tells  $\mu$ Torrent to override **the default queueing settings** for the selected torrent job(s).
- The **Minimum ratio** field behaves identically to the respective option in the default queueing preferences, but affects only the torrent job(s) whose properties are being edited.
- The **Minimum seeding time** field behaves identically to the respective option in the default queueing preferences, but affects only the torrent job(s) whose properties are being edited.
- The **Minimum number of available seeds** field behaves identically to the respective option in the default queueing preferences, but affects only the torrent job(s) whose properties are being edited..

## Other Settings

- **Initial Seeding** enables a more bandwidth efficient method of seeding the torrent job(s). This should be used only if you are the sole seeder in the swarm, and if there are at least 2 peers connected. Generally, this option should not be used by people with high upload speeds.
- **Enable DHT** enables announcing the torrent over DHT. This option is automatically disabled if DHT is turned off, or if the selected .torrent file has the 'private' flag set.
- **Peer Exchange** enables trading of peer lists with other peers that support  $\mu$ Torrent's implementation of Peer Exchange. This option is disabled if the selected .torrent file has the 'private' flag set.
- **Local Peer Discovery** enables discovery of local peers via multicast. This option is disabled if the selected .torrent file has the 'private' flag set.



[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Miscellaneous](#) > [Torrent Properties](#)

# Advanced

## Web Seeds

- The edit control lists all the web seeds the selected torrent job(s) will use, and is interpreted identically to how the "Create New Torrent" dialog's "Web Seeds" edit control is interpreted.

## Run Program

- The given field allows you to run a program after the selected torrent job(s) finish downloading. There are several variables you should be aware of if you wish to use this feature to manipulate the files you downloaded:
  - **%F** is the placeholder for the name of the file in the torrent job. This variable assumes the selected torrent job does not contain multiple files. If this is not the case, it will be replaced with an empty string (nothing).
  - **%D** is equivalent to the path shown in the **Save As** field, assuming the torrent job contains multiple files. If it is a single-file torrent job, the path to the directory containing the file is returned.
  - **%P** - is the previous state of torrent
  - **%N** is the same as what's listed under the **Name** column for the torrent job.
  - **%S** triggers the execution of the program when the torrent state changes to it (see below).
  - **%L** is the placeholder for the label that is assigned to this torrent
  - **%T** is the placeholder for the tracker(s) used with this torrent
  - **%M** is the placeholder for the status message string in the status column
  - **%I** is the placeholder for the hex encoded info hash
  - **%S** is the state of torrent
  - **%K** is the placeholder for kind of that torrent - with a single file of multiple files.

State: is a combination of the following:

- 1 - started
- 2 - checking
- 4 - start-after-check

- 8 - checked
- 16 - error
- 32 - paused
- 64 - auto
- 128 - loaded

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >

[Miscellaneous](#)

# Tray Icon

The  $\mu$ Torrent tray icon gives you a brief overview of your torrent jobs. Placing the mouse cursor over the tray icon will show a tooltip where you can view the total number of torrent jobs you have currently downloading, the total amount queued for download, the total amount currently seeding, and the total amount queued for seeding. Also, the total upload and download speeds are displayed in the tooltip. **Left-clicking** on the tray icon will toggle the visibility of the main window, depending your **system tray settings**. **Right-clicking** the tray icon will display a context menu, through which you can make quick settings changes to  $\mu$ Torrent:

- **Hide/Show  $\mu$ Torrent** toggles the display of the  $\mu$ Torrent main window.

---

- **Pause all torrents** sets all downloading or seeding torrent jobs to paused mode.
- **Resume all torrents** will resume the transferring of paused torrent jobs.

---

- **Download Limit** allows you to control the global maximum download limit. The values displayed depend on **the speed popup list settings** in the Preferences.
- **Upload Limit** allows you to control the global maximum upload limit. The values displayed depend on **the speed popup list settings** in the Preferences.
- **Enable Scheduler** allows you to enable or disable the scheduler from the without having to open the Preferences.

---

- **$\mu$ Torrent Webpage** opens the  **$\mu$ Torrent webpage** in your default

web browser.

- **µTorrent Forums** opens the [µTorrent forums](#) in your default web browser.
- 

- **Exit** does just that -- it exits µTorrent. Note that exiting µTorrent automatically sends a stop signal to trackers, so you do not have to stop all torrent jobs before exiting. Torrent jobs that are not stopped will automatically be started when µTorrent is next opened. Even after exiting, the µTorrent process might continue to run for a while longer. This happens because µTorrent is trying to finish transferring pieces and write remaining pieces to disk from memory. The process should disappear after a few seconds.

[« Previous](#)   [Next »](#)

µTorrent User Manual

## Appendix B: Advanced Information

This appendix provides advanced information pertaining to operation of  $\mu$ Torrent.

- [Settings Directory](#)
- [URL Formatting](#)
- [Wildcards](#)
- [Command Line Options](#)
- [Keyboard Shortcuts](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix B: Advanced Information](#)

## Settings Directory

Although  $\mu$ Torrent is mostly self-contained, there has to be a location where it can store its settings for later use after the first run. Because  $\mu$ Torrent does not use the Windows Registry, it has to store its configuration in files on separate files on the disk. By default, those files are located at %AppData%\uTorrent, which can be easily accessed by visiting "Start" > "Run" (or by pressing **Windows** + **R** on the keyboard), typing it into the "Open" field, and pressing "OK." Typically, you'll find several .dat files in the settings directory:

- **dht.dat** contains information regarding DHT that  $\mu$ Torrent uses when connecting to the DHT network.
- **resume.dat** contains information regarding currently loaded torrent jobs.
- **rss.dat** stores all RSS-related settings, and also holds the history of previously downloaded torrent jobs. This file only gets created if you use the **RSS Downloader**.
- **settings.dat** contains most of the settings in  $\mu$ Torrent, and also contains the information listed in the statistics dialog.

Note that it is normal to find .dat.old and .dat.\*.bad files in the same directory. The former file type denotes good backup files for the case that the corresponding .dat file becomes corrupted. The latter file type denotes backups of corrupted files, which may be sent to the developer for analysis if necessary.

In addition to .dat, .dat.old, and .dat.\*.bad files,  $\mu$ Torrent stores loaded .torrent files in its settings directory unless **a storage location** is specified. These files need to be stored in order for  $\mu$ Torrent to work properly, so do not delete them unless you are absolutely sure  $\mu$ Torrent no longer has the associated torrent job loaded any longer. If you want  $\mu$ Torrent to delete the stored .torrent files automatically upon removal of the torrent job, see the description of the **Remove** button in the toolbar.

## Encapsulated "Installation"

While %AppData%\uTorrent is the default directory for storing the settings,  $\mu$ Torrent actually searches the directory that the executable file is located in first before looking in %AppData%\uTorrent. As such, so long as the directory that the executable is located in contains a settings.dat file (even a blank one),  $\mu$ Torrent will look no further, and use only the files located in its current directory instead. With this behavior, it is possible to create encapsulated  $\mu$ Torrent "installations" for use on portable flash drives, or just to have an isolated copy for any other reason.

If you don't already have an encapsulated install, you can manually move the contents of %AppData%\uTorrent into the same directory as the  $\mu$ Torrent executable to create such an install.

## External Files

Besides the default .dat files,  $\mu$ Torrent can make use of several other files external from the executable. Such files may modify several subtle behaviors in  $\mu$ Torrent, or allow you to customize the graphical look and feel of  $\mu$ Torrent. What all external files have in common is the location at which they should be placed -- the settings directory.

## Extended Functionality

- **ipfilter.dat** is a simple text file that specifies IP ranges to block. Only blocks are supported; any allows will be ignored. The format is as follows:
  - **IPv4:** xxx.xxx.xxx.xxx - yyy.yyy.yyy.yyy for each line, where each x and each y corresponds to arbitrary decimal values between 0 and 9.
  - **IPv6:** [xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx] - [yyyy:yyyy:yyyy:yyyy:yyyy:yyyy:yyyy:yyyy] for each line, where each x and each y corresponds to arbitrary hexadecimal values between 0 and F. Zero-grouped IPv6 addresses are accepted.

Single IPs can be specified on a line without having to write it in IP range notation. This file can be reloaded via the [context menu in the Peers tab](#). If you wish to make use of this file, [ipfilter.enable](#) should

be enabled. Note that only peer connections are blocked by `ipfilter.dat`; tracker connections (including DHT) are not blocked, even if their IPs fall within any ranges specified in `ipfilter.dat`.

- **utorrent.chm** is the user manual. The manual can be accessed via the [help entry in the Help menu](#), or by pressing **F1** on the keyboard.
- **utorrent.lng** is the file that contains all available translations for  $\mu$ Torrent at the time of download. When first added to the settings directory,  $\mu$ Torrent will automatically switch to the system language if a translation is available. The [language can be changed](#) in the Preferences.
- **webui.zip** is a zip file containing data  $\mu$ Torrent will serve when the [Web UI](#) is enabled and you access `http://IP:port/gui/` in a web browser, where IP is your computer's IP address, and port is the port  $\mu$ Torrent is listening on. Using Web UI, you can control  $\mu$ Torrent from any other computer with an Internet connection and a compatible browser installed.

## Interface Customization

- **flags.conf** & **flags.bmp** are files used to replace the internal  $\mu$ Torrent assignment of flags to a peer's resolved IP's host domain. `flags.conf` is a text file that specifies the flag order, and maps host domains to a country, whose flag is then obtained from `flags.bmp`. `flags.conf` only works on hosts with a TLD of `.net` or `.com`, each mapping in the format `host domain|country code`, where the country code is the country's IANA-assigned two-letter code. `flags.bmp` is a Windows bitmap that contains adjacently-conjoined bitmaps of dimensions 16x16 each (without any space in between), each 16x16 square being a flag. A custom `flags.conf` and `flags.bmp` is maintained as [a community project in the  \$\mu\$ Torrent forums](#), compiled by eng. Pressing **Ctrl** + **Shift** + **R** on your keyboard tells  $\mu$ Torrent to (re)load `flags.conf`. If [peer.resolve\\_country](#) is enabled, these files will be ignored.
- **main.ico** is the icon that will be displayed in the icon at the top-left

of the  $\mu$ Torrent window.

- **maindoc.ico** is the icon that will be used as a file type icon for .torrent files **upon association**. This is mainly used for older versions of Windows, which might have problems generating file type icons automatically for .torrent files upon association.
- **tray.ico** is the icon that will be displayed in the system tray if the relevant **system tray settings** are enabled.
- **tabs.bmp** is a Windows bitmap of dimensions 224x16, consisting of 14 adjacently-conjoined bitmaps of dimensions 16x16 each (without any space in between), that  $\mu$ Torrent uses **when icons are displayed on detailed info pane tabs**. Icons are arranged in the following order:
  - General
  - Trackers
  - Peers
  - Pieces
  - Files
  - Speed
  - Logger
  - General (inactive)
  - Trackers (inactive)
  - Peers (inactive)
  - Pieces (inactive)
  - Files (inactive)
  - Speed (inactive)
  - Logger (inactive)
- **toolbar.bmp** is a Windows bitmap of dimensions 288x24, consisting of 12 adjacently-conjoined bitmaps of dimensions 24x24 each (without any space in between), that  $\mu$ Torrent uses for the images in **the  $\mu$ Torrent toolbar**. Icons are arranged in the following order:
  - Add Torrent
  - Add Torrent from URL
  - Create New Torrent
  - Remove
  - Start

- Pause
  - Stop
  - Move Up Queue
  - Move Down Queue
  - Search
  - Add RSS Feed
  - Preferences
- **tstatus.bmp** is a Windows bitmap of dimensions 304x16, consisting of 19 adjacently-conjoined bitmaps of dimensions 16x16 each (without any space in between), that  $\mu$ Torrent uses for the images in [torrent job status icons](#) and [the category list](#). Icons are arranged in the following order:
    - Downloading
    - Seeding
    - Stopped
    - Paused
    - Downloading (tracker error)
    - Seeding (tracker error)
    - Critical Error
    - Stopped and Finished
    - Queued Download
    - Queued Seed
    - Active (category icon)
    - All (category icon)
    - Inactive (category icon)
    - RSS Feed (icon 1)
    - RSS Feed (icon 2)
    - RSS Feed (icon 3)
    - RSS Feed (invalid feed)
    - RSS Item (new release, within 24 hours)
    - RSS Item (downloaded)

Multiple RSS Feed icons are needed because  $\mu$ Torrent treats the icons as separate frames in an animation to indicate that there is RSS activity.

External files that change the  $\mu$ Torrent interface are numerous, but

thankfully, there is a central repository where you can find many of them. Check out the [µTorrent Skins Page](#) for your skinning needs! If you ever decide to create your own interface files, feel free to submit them on that page, and inform us about it in the forums!

---

« Previous    Next »

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix B: Advanced Information](#)

# URL Formatting

In most cases, when entering a URL into  $\mu$ Torrent, the usual `http://domain/path` works just fine, but in certain special cases where the site is protected by some form of authentication, modifications will have to be made to the URL in order for  $\mu$ Torrent to be able to get to the contents.

Sites that require HTTP authentication will have to be entered as such: `http://username:password@domain/path`

Sites that require cookies must be entered as such: `http://domain/path:COOKIE:uid=U;pass=P`

Note that U and P must be obtained from the cookie file for the site. Also, realize that some sites do not use `uid` and/or `pass` as the corresponding variables, and as such, you must use the exact variable name and extra variables they do specify. For example, if the site's cookies specify `a`, `b`, and `c`, with values `A`, `B`, and `C` respectively, you should enter the URL as such: `http://domain/path:COOKIE:a=A;b=B;c=C`

- Internet Explorer users can find their cookies in `%UserProfile%\Cookies`
- Chrome users can find cookies in "*Chrome Menu*" > "*Settings*" > "*Show Advanced Settings*" > "*Content settings*" > "*Cookies*"
- Firefox users can find their cookies in "*Tools*" > "*Options*" > "*Privacy*" > "*Cookies*" > "*Show Cookies*"
- Opera users can find their cookies in "*Tools*" > "*Preferences*" > "*Advanced*" > "*Cookies*" > "*Manage cookies...*"
- Users of other web browsers will have to consult their respective documentation

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix B: Advanced Information](#)

# Wildcards

In several places in  $\mu$ Torrent, wildcards can be used in place of normal characters in order to specify a pattern of characters that  $\mu$ Torrent should use to match whatever it needs. The following may be used in those situations:

- \* (asterisk) to match any text of any length
- ? (question mark) to match any single character
- | (vertical pipe) is an "or" conditional

**Example:** `*File?A*|File?B*` matches any file that contains `File?A` anywhere in the filename (where ? is any single character, such as a space or an underscore), or any file that begins with `File?B` in its filename (again, where ? is any single character).

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix B: Advanced Information](#)

# Command Line Options

There are several command line options you can use with  $\mu$ Torrent to make it behave differently than normal. Do note that these options can only be used when  $\mu$ Torrent is opened through a shortcut, by command line, or some other means that allows command line options to be used. At any time, a space should be appended along with the wanted option at the end of the target path for the shortcut, outside of any quotation marks that may exist in the target path.

- **/ANSI** starts  $\mu$ Torrent without Unicode support, in case anyone ever has trouble with Unicode.
- **/AUTOUPDATE** "FILE PATH" [/NORUN] replaces the file located at FILE PATH with the currently-running executable and starts the executable from the specified path. If **/NORUN** is used in conjunction with this option, then  $\mu$ Torrent won't run the executable after the replacement occurs.
- **/BIGDUMP** tells  $\mu$ Torrent to create a large memory dump instead of the minidumps it normally creates when it crashes. This can be useful for the developer to debug problems where minidumps provide insufficient information.
- **/BRINGTOFRONT** forces the  $\mu$ Torrent window to be shown, regardless of whether it was previously minimized on exit, or whether **/HIDE** or **/MINIMIZED** were also used as command line options.
- **/DIRECTORY** "SAVE PATH" ".TORRENT FILE TO OPEN" allows you to open a .torrent file and specify where you'd like to save the torrent contents to from the command line (without any GUI interaction), where "SAVE PATH" is the path to the location you'd like to save the torrent contents, and ".TORRENT FILE TO OPEN" is the location of the .torrent file. Note that the save path must not contain a trailing backslash, otherwise  $\mu$ Torrent will fail to load the

.torrent file. Only local .torrent files may be used.

- **/HIDE** starts  $\mu$ Torrent in boss-key mode, meaning the tray icon and the entire interface is hidden until the boss-key is pressed. Note that you should **set up a boss-key** before actually starting  $\mu$ Torrent with this switch, otherwise you might find yourself having a difficult time seeing and using  $\mu$ Torrent.
- **/LAUNCHBUNDLEURL** "URL" opens URL in your default web browser.
- **/LOGFILE** "SAVE PATH" will perform the initial file selection for the **log to file** feature automatically on startup.
- **/MINIMIZED** will start  $\mu$ Torrent minimized, and if **minimize to tray** is enabled, will start  $\mu$ Torrent minimized to tray.
- **/NOINSTALL** will tell  $\mu$ Torrent to bypass the installation dialog prompt.
- **/PERFORMINSTALL** FLAGS is a bitfield that tells  $\mu$ Torrent what shortcuts to create, depending on the following possible values for FLAGS:
  - **2** tells  $\mu$ Torrent to create a shortcut in the Start menu
  - **4** tells  $\mu$ Torrent to create a shortcut on the Desktop
  - **8** tells  $\mu$ Torrent to create a shortcut in the Quick Launch toolbar

Because the flags are interpreted as a bitfield, values can be added together to perform each of the combined actions.

- **/RECOVER** will allow you to open up a second instance of  $\mu$ Torrent instead of its normal behavior where it simply activates the currently-running instance. In order to prevent possible problems, it is recommended that you create an **encapsulated copy** of  $\mu$ Torrent before opening it as a second instance, then create a shortcut for the encapsulated copy of the executable with the **/RECOVER** option. Remember that two applications cannot listen on the same port, so if you intend on having the second instance actually be a properly-working instance, you should set the second instance to

listen on a different port, and forward that port as well.

- **/UNINSTALL** [*S*] will uninstall  $\mu$ Torrent after user confirmation, deleting even the **settings directory**. If the install is broken enough that this switch doesn't work, running this option in conjunction with the ***IS*** option should force an uninstall without asking for confirmation.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > Appendix B: Advanced Information](#)

# Keyboard Shortcuts

Here is a consolidated list of the keyboard shortcuts in  $\mu$ Torrent:

COMMAND	SHORTCUT
Add Torrent	Ctrl + O
Add Torrent (no default save)	Ctrl + D
Add Torrent from URL	Ctrl + U
Create New Torrent	Ctrl + N
Torrent Job Manipulation	
Move Down Queue	Ctrl + Alt + Down
Move Up Queue	Ctrl + Alt + Up
Remove	Delete
Remove and delete Data	Shift + Delete
Rename	F2
Filter Torrents	Ctrl + F
Dialogs	
Preferences	Ctrl + P
RSS Downloader	Ctrl + R
Setup Guide	Ctrl + G
Interface Layout	
Show Category List	F7
Show Detailed Info	F5
Show Status Bar	F6
Show Toolbar	F4
RSS Downloader	
Remove RSS History Entry	Delete
Remove RSS Favorite	Delete

Rename RSS Favorite	F2
Miscellaneous	
µTorrent Help	F1
Copy Selected Logger Tab Text	Ctrl + C
Copy Statistics Dialog Text	Ctrl + C
Reload flags.bmp / flags.conf	Ctrl + Shift + R
Remove Tracker (from Trackers list)	Delete
« Previous    Next »	

[« Previous](#)   [Next »](#)

µTorrent User Manual

# FAQ: Frequently Asked Questions

This section of the manual attempts to answer questions many people might ask about  $\mu$ Torrent.

- [Installation](#)
- [Features](#)
- [Usage](#)
- [Network](#)
- [Troubleshooting](#)
- [Error Messages](#)
- [Incompatibilities](#)
- [Miscellaneous](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual > FAQ: Frequently Asked Questions](#)

# Installation

## Does $\mu$ Torrent install itself?

$\mu$ Torrent will install itself only if the user chooses to install it when prompted to the first time it is run. Installation is not mandatory, and  $\mu$ Torrent can still run properly without the need to be installed.

## How do I backup my settings?

Simply make a copy of your `settings directory`.

## How do I make $\mu$ Torrent self-contained in one directory?

Create an `encapsulated installation`.

## How do I reset the settings back to the defaults?

Simply delete `settings.dat` and `settings.dat.old` from your `settings directory` while  $\mu$ Torrent is not running.

## How do I share my torrents between multiple users?

Create an `encapsulated installation` of  $\mu$ Torrent and make sure you set the directory and all of its contents to be readable and writable by other users.

## How do I uninstall $\mu$ Torrent?

If you allowed  $\mu$ Torrent to create an uninstall entry in the Windows

Registry during the initial installation dialog, then all you need to do is uninstall it from the "*Add or Remove Programs*" applet in the Control Panel. Otherwise, you can simply delete the  $\mu$ Torrent executable file from wherever it is stored. If you are concerned about the settings being left behind on your computer, delete your entire **settings directory** as well.

## How do I use $\mu$ Torrent on a USB key or some other removable drive?

Create an **encapsulated installation** of  $\mu$ Torrent on the target device.

## Where are the settings and .torrent files stored?

The files are stored in your **settings directory**.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > FAQ: Frequently Asked Questions](#)

# Features

## Does µTorrent have an endgame mode?

Yes, µTorrent automatically enters endgame mode when the pieces remaining are all being actively downloaded. During endgame mode, µTorrent requests each remaining piece from multiple peers instead of requesting each piece from a single peer. This mode helps make getting the last pieces of the torrent contents take much less time than it normally would.

## Does µTorrent have Unicode support?

Yes, µTorrent supports Unicode.

## Does µTorrent support HTTPS or UDP trackers?

Yes, µTorrent supports HTTPS (SSL) for both trackers and RSS feeds. UDP trackers are supported as well.

## Does µTorrent support magnet URIs?

Yes, µTorrent can use and create magnet URIs compatible with the Azureus magnet URI implementation. µTorrent can [generate magnet URIs](#) for any torrent job, and magnet URIs can be opened from the [Add Torrent from URL](#) dialog. Metadata regarding the torrent contents are obtained from other peers that support the feature.

## Does µTorrent support multi-scrape?

Yes, µTorrent supports multi-scrape and uses it when [bt.multiscrape](#) is enabled. It automatically detects trackers that do not support multi-

scrape and falls back to single-scrape mode for those trackers.

## Does $\mu$ Torrent support multi-tracker .torrent files?

Yes,  $\mu$ Torrent supports multi-tracker .torrent files.  $\mu$ Torrent announces to all tracker tiers simultaneously, but only one tracker per tier. To see all the trackers in the .torrent file, simply check the [torrent job's Properties](#). To learn more about multi-tracker specification, check the [multitracker specifications](#) at Depthstrike.com's wiki.

## Does $\mu$ Torrent support trackerless .torrent files?

Yes,  $\mu$ Torrent supports trackerless .torrent files. The built-in .torrent file creator can also create trackerless .torrent files.

## Does $\mu$ Torrent support UNC-style paths or network drives?

Yes,  $\mu$ Torrent supports UNC-style paths (like \\192.168.1.2\C\$\). Note that there may be increased disk fragmentation when saving to a network drive, since pre-allocation of files is not possible across the network.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > FAQ: Frequently Asked Questions](#)

# Usage

## How do I add additional columns to a list-view?

You can select what columns you wish to see in a list-view by [right-clicking](#) the column headers. If you wish to reset all changes you've made to the columns, you can select "Reset" in that context menu.

## How do I block specific peers from connecting to me?

You can create an [ipfilter.dat](#) file and load it in  $\mu$ Torrent, and the specified peers will be blocked.

## How do I change the country flags shown in the Peers tab?

You can use [flags.conf](#) and [flags.bmp](#) to customize the country flags.

## How do I change the icons in $\mu$ Torrent?

You can change the icons by placing [the respective interface customization file](#) in the  $\mu$ Torrent settings directory in order to customize the interface. A restart is required in order for new interface icons to be loaded.

## How do I change the interface language?

You can install [utorrent.lng](#) (by having  $\mu$ Torrent [download the translation file](#), or by getting a copy from the [download page](#) yourself).

Afterwards, [select the language](#) you wish to use from the general preferences.

## How do I configure the disk cache?

The most important step is to understand how [the disk cache options](#) affect  $\mu$ Torrent. Options that increase memory usage decrease disk access, while options that decrease memory usage increase disk access.

## How do I create my own .torrent file?

You can use the [Create New Torrent](#) feature in  $\mu$ Torrent to create your own .torrent file to share with other people.

## How do I enable DHT?

You can enable it from [the BitTorrent settings](#). Although DHT is enabled, not all torrent jobs might use it, as .torrent files with the 'private' flag set won't announce to the DHT network. Additionally, any torrent job that does not [have DHT enabled](#) won't be announced on DHT.

## How do I enable Initial Seeding (Super Seeding)?

You can enable it from [the torrent job properties](#). If you are not the only seeder in the swarm, you should not enable Initial Seeding.

## How do I enable Local Peer Discovery?

Local Peer Discovery is enabled on a per torrent job basis. You can enable it from [the torrent job properties](#), assuming the .torrent file does not have the 'private' flag set. To automatically enable Local Peer Discovery for any newly added torrent job, check the [Enable Local Peer](#)

Discovery option in the Preferences.

## How do I enable Peer Exchange?

Peer Exchange is enabled on a per torrent job basis. You can enable it from [the torrent job properties](#), assuming the .torrent file does not have the 'private' flag set. To automatically enable Peer Exchange for any newly added torrent job, check the [Enable Peer Exchange](#) option in the Preferences.

## How do I enable Protocol Encryption?

Enable [Protocol Encryption](#) in the Preferences. If your ISP is known to throttle or block BitTorrent traffic, and you find that enabling Protocol Encryption is not helping, then you may want to consider setting the encryption to "*Forced*" and disable legacy incoming connections.

## How do I enable NAT Port Mapping Protocol (NAT-PMP)?

You can enable it from [the connection preferences](#). Note that not all routers support NAT-PMP, in which case you will have to [forward your port manually](#).

## How do I enable Universal Plug and Play (UPnP)?

You can enable it from [the connection preferences](#). Note that not all routers support UPnP, in which case you will have to [forward your port manually](#).

## How do I load a .torrent file from the command line?

Run the  $\mu$ Torrent executable with the [/DIRECTORY](#) command line

option.

## How do I make $\mu$ Torrent allocate all the files when I start a torrent job?

Enable `Pre-allocate all files` in the general Preferences.

## How do I make $\mu$ Torrent append a !ut extension to incomplete files?

Enable `Append !ut to incomplete files` in the general Preferences. The option takes effect immediately after the changes to the Preferences are saved.

## How do I make $\mu$ Torrent auto-load .torrent files from a specified directory?

Enable the `auto-load option` in the Preferences and select the directory you would like to automatically load .torrent files from.

## How do I make $\mu$ Torrent automatically ban peers after a certain number of hashfails?

`bt.ban_threshold` determines the number of hashfails  $\mu$ Torrent takes action against it. If `bt.use_ban_ratio` is disabled,  $\mu$ Torrent will ban the peer immediately after it passes the ban threshold. Otherwise,  $\mu$ Torrent will look at the ratio of good to bad pieces that the peer has sent, and if it finds that the ratio is lower than `bt.ban_ratio`, it will then ban the peer. If enough peers in the same range are banned, and `bt.use_rangeblock` is enabled,  $\mu$ Torrent will ban the entire range, since there is a good probability that the entire range is poisoning BitTorrent swarms.

## How do I make µTorrent automatically move files when a torrent job finishes?

You can enable the **Move completed downloads to** option in the Preferences and specify the directory you wish to move finished torrent jobs to. If you want to organize the selected directory by torrent job label, enable the **Append the torrents label** option.

## How do I make µTorrent automatically run a program after a download finishes?

In the torrent job's properties, there is a **Run Program** section in the advanced section. When properly filled out, µTorrent will run the selected application immediately after the torrent job finishes downloading.

## How do I make µTorrent delete files to the Recycle Bin?

Enable the **Move to trash if possible** option in the Remove button's context menu in the toolbar. Note that files larger than the Recycle Bin's capacity will be removed from the drive immediately (bypassing the Recycle Bin entirely).

## How do I make µTorrent download files from an RSS feed?

The first step is to find a RSS feed that links to .torrent files in at least one of either its <guid>, <link>, or <enclosure> tags. After you **add such a feed** (making sure to provide the authentication information if necessary), you can create a **Favorites filter** for the feed that automatically downloads and loads the linked .torrent files from the feed that match the filter you specify. Make sure you familiarize yourself with all of the filter options, as they can be confusing. When the filter is set up properly and is enabled, µTorrent will take care of the downloading for

you.

## How do I make $\mu$ Torrent prioritize the first and last piece of each file?

Enable `bt.prio_first_last_piece` in the advanced Preferences.

## How do I make $\mu$ Torrent start in boss-key mode?

Run  $\mu$ Torrent with the `/HIDE` command line option.

## How do I make $\mu$ Torrent start minimized?

Run  $\mu$ Torrent with the `/MINIMIZED` command line option.

## How do I make $\mu$ Torrent stop downloading and only upload?

To perform this on any and all torrent jobs, set the scheduler to `seeding only` mode during the times you want it to upload only.

To perform this on a specific torrent job, stop the torrent job, select all of its files in the `Files tab`, then select "*Don't Download*."  $\mu$ Torrent might continue to download pieces if there are any left incomplete (you can check the `Pieces tab`), but afterwards, it won't request any new pieces to download from other peers.

## How do I make $\mu$ Torrent stop seeding a torrent job at a specific share ratio?

Set the `Limit the upload rate to` option to `0`. Now, whenever any torrent job reaches the seeding goal, it will be stopped.

## How do I make µTorrent use my proxy?

You can fill in the proxy information in [the Connection preferences](#). Unless you are sure your proxy server can handle the load, you should refrain from [using the proxy server for peer-to-peer connections](#).

## How do I manually announce to trackers?

The [Update Tracker](#) feature in the torrent job context menu can be used to update all trackers in the selected torrent job(s). Alternatively, specific trackers can be updated via the [Trackers tab context menu](#). Although an anti-hammer feature has been put in place to prevent abuse of the feature, manual update of the tracker should still be avoided where not absolutely necessary in order to minimize strain on trackers.

## How do I move a torrent job to the top or bottom of the queue?

Hold [Shift](#) on your keyboard while pressing the [Move Up Queue](#) toolbar button to move the selected torrent job(s) to the top of the queue. Hold [Shift](#) on your keyboard while pressing the [Move Down Queue](#) toolbar button to move the selected torrent job(s) to the bottom of the queue. The respective move up/down queue items from the [torrent jobs list context menu](#) behave in the same way when [Shift](#) is held while the menu item is selected.

## How do I quickly change the upload and download limits?

The global transfer rate limits can be quickly set from the [status bar context menu](#). If visible, the global transfer rate limits can also be quickly controlled from [the system tray context menu](#).

For to quickly change the transfer rate limits of specific torrent jobs, select the torrent jobs you would like to modify, then [right-click](#) it and

use the [Bandwidth Allocation](#) menu.

## How do I rename a torrent job in the torrent jobs list?

A torrent job can be renamed in the list by slow [double-clicking](#) on the torrent job, or by pressing [F2](#) on the keyboard while the torrent job is selected. To cancel the renaming, you should press [Esc](#) on the keyboard. To confirm the change, you should press [Enter](#) on the keyboard, or use the mouse to click anywhere else on the torrent jobs list.

## How do I rename a torrent job's download directory?

In the [Add Torrent Jobs dialog](#), you can edit the target directory before confirming the save location. If you wish to rename or move an existing torrent job's current location on disk:

1. Stop the torrent job whose contents you wish to move.
2. Rename or move the torrent contents.
3. Use [Set Download Location...](#) from the torrent jobs list context menu to select the new location.
4. If  $\mu$ Torrent doesn't automatically recheck the file, and you wish to confirm the integrity of the files, [force a re-check](#) on the torrent job.

## How do I run multiple instances of $\mu$ Torrent simultaneously?

To run multiple instances of  $\mu$ Torrent, you can run each new instance with the [/RECOVER](#) command line option. Because of the possibility of conflicts, you should perform the following before running multiple instances:

1. Create a new folder containing an **encapsulated installations** of  $\mu$ Torrent (you should copy the executable).
2. Create a shortcut for this copy of the executable.
3. Edit the shortcut's target by adding `/RECOVER` to the end of the "*Target*" path, outside of any quotation marks, and separated from the executable path by a space.
4. Run the shortcut to open a new instance of  $\mu$ Torrent.
5. Configure this instance of  $\mu$ Torrent to use a different **listening port** (and different **alternative listening port** if applicable).
6. If applicable, **forward the port(s)** for this instance in your router and firewall.

Because bandwidth must be split for each instance, you should probably run **the Setup Guide** again for each instance, but select appropriate portions of your connection's maximum upload rate for each instance you run.

---

## How do I set a boss-key up?

A boss-key can be set from **the privacy preferences**. After setting it and confirming the changes,  $\mu$ Torrent's visibility can be toggled by pressing the selected key combination.

---

## How do I set the default torrent job Remove action?

The default Remove action can be set by **right-click**ing the **Remove** button in the toolbar, holding **Shift** on the keyboard, then **left-click**ing one of the four possible options. Alternatively, the default behavior can be configured by setting `gui.default_del_action` appropriately.

---

## How do I set the download location for torrent jobs added via magnet URI?

With magnet URIs,  $\mu$ Torrent does not have a copy of the info dictionary on hand, so it does not know the name or the contents.

Consequently, it must wait to find a source with the dictionary, which may sometimes take a long time. Because of this, µTorrent uses [the default download location](#) as the destination for the files (in case it happens to find the info dictionary when the user is not available to set the download location manually). If no default download location is set, µTorrent will use %UserProfile%\My Documents\Downloads as the destination.

## How do I skip certain files in a torrent job?

Files can be skipped in the [Files list in the Add New Torrent dialog](#). Alternatively, files can be skipped via the [Files tab context menu](#).

## How do I sort by more than one column in a list-view?

You can sort by any column by [left-clicking](#) on its header. To sort on another column as a secondary sort order, hold [Shift](#) on the keyboard while [left-clicking](#) on the column header you wish to use as the secondary sort criterion.

## How do I use the embedded tracker?

Enable [bt.enable\\_tracker](#) in the advanced preferences. Make sure you carefully read the description of the option.

## How do I use the scheduler to control transfer rates?

The most important step is to understand how [the scheduler options](#) affect µTorrent. The rest is as simple as clicking on the time slot you wish to set up to the color corresponding to the effect you want for the time slot.

## How do I use the Web UI?

Enable the **Web UI**, and  $\mu$ Torrent should automatically install it for you. Afterwards, fill in the username and password you would like to use to log into the web interface. If you wish to use a different listening port from the **normal listening port** for the web interface, specify it in the **Alternative listening port** field. Additionally, if you wish to allow only specific IP address to access your web interface, fill in the **restriction field** accordingly.

## What are labels and what can they be used for?

Labels are used as a powerful tool for torrent job organization in  $\mu$ Torrent. They can make torrent job identification and sorting much easier for you.

To set a label for a torrent job, use the **Label** submenu in the torrent job's context menu. A benefit of using labels is that they can also be used to organize torrent contents (not just torrent jobs) with the **Move completed downloads to** option.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > FAQ: Frequently Asked Questions](#)

## Network

### Does µTorrent work well on Windows XP SP2 or higher operating systems with an unpatched TCPIP.sys?

Yes, by default, µTorrent will make at most 8 simultaneous connection attempts to work within the 10 connection attempt (half-open connection) limit on these operating systems.

Patching TCPIP.sys to a higher limit may help if you are having problems with your Internet connection, though increasing the limit may also cause some routers to freeze up (due to the increased rate of simultaneous connection attempts). If you would still like to patch the file, download the [EventID 4226 Patcher from LvlLord.de](#).

Beware that in most cases, you should refrain from setting the TCPIP.sys limit to any number greater than 50, as there are few (if any) benefits to be gained from higher numbers. Also understand that [net.max\\_halfopen](#) should always be set to a number *lower* than the TCPIP.sys limit set by the patcher (at most 80% of the TCPIP.sys limit only). For more information on how (or whether) the half-open connection limit should be set, see [this FAQ entry](#).

Microsoft has made a habit of reverting the TCPIP.sys connection attempt limit through Windows Update, which (for many users) occurs on a monthly basis. If you decide to patch your TCPIP.sys file and raise the [net.max\\_halfopen](#) limit along with it, then remember to repatch the file every time you perform a Windows Update (otherwise, you'll most likely run into connection troubles).

### How can I tell if a peer is an incoming or outgoing

## connection?

You can check the [Flag column in the Peers tab](#) for the peer.

## How do I change the number of active torrents jobs or downloads allowed to run simultaneously?

Change the [queue settings](#). It is recommended that you not modify this setting unless you understand very clearly what the consequences are behind changing this setting. Starting too many torrent jobs simultaneously can be detrimental to bandwidth usage, so if you are not certain about your changes, stick to the numbers recommended by [the Setup Guide](#).

## How do I change the number of connections µTorrent uses?

Set the relevant options in [the preferences](#). It is recommended that you not modify this setting unless you understand very clearly what the consequences are behind changing this setting. Starting too many torrent jobs simultaneously can be detrimental to bandwidth usage, so if you are not certain about your changes, stick to the numbers recommended by [the Setup Guide](#).

## How do I change the port µTorrent uses?

The listening port for incoming connections can be set with the [Port used for incoming connections](#) option in the Preferences. For outgoing connections, µTorrent uses the ephemeral port range by default, but this can be changed by using the [net.outgoing\\_port](#) and (optionally) [net.outgoing\\_max\\_port](#) options.

## How do I forward ports?

Read the [Port Forwarding guide](#).

## How do I hide my IP address?

You don't. BitTorrent wasn't designed with anonymity in mind. Because it requires that peers know each other's IP addresses to transfer with one another, spoofing or hiding your IP address won't do you much good with regards to speed and/or connectivity.

VPN or proxy services may allow you to masquerade your IP address behind other IPs, but because they act as middlemen, you are effectively firewalled behind such services, which is [a bad thing](#).

## How do I make $\mu$ Torrent report a different IP to the tracker? I'm behind a proxy and need this function.

Set the [IP/Hostname to report to tracker](#) option in the Preferences.

## How do I make $\mu$ Torrent use a different upload speed when seeding?

Set the [Alternate upload rate when not downloading](#) option in the Preferences.

## How do I make $\mu$ Torrent use a specific network adapter?

Set the [net.bind\\_ip](#) option to the IP address of the adapter you wish to use for incoming connections. Set the [net.outgoing\\_ip](#) option to the IP address of the adapter you wish to use for outgoing connections.

## What port should I use for $\mu$ Torrent?

It is generally recommended that you not pick ports in the 6881-6889 range, as they are commonly throttled by ISPs. Since no single port has inherent advantages over any other port, you can simply let  $\mu$ Torrent **pick a random port** for you.

## What should I set my half-open connection limit to?

To answer this question, a bit of background information may be required. The "half-open" (technically, "embryonic") connection limit controls how many connections  $\mu$ Torrent will attempt to establish simultaneously at any given time. Half-open connections are just like phone calls that haven't yet been picked up by the other end of the line. The half-open connection limit controls the number of such "calls" you can attempt to make at any given time, but does not limit how many fully-established connections (phone calls that are successfully picked up from the other end) you can make in total.

An important point to note (that is commonly misunderstood and misrepresented) is the fact that the half-open connection limit does not affect the overall speed at which you can download or upload. At best, increasing the half-open limit may decrease the amount of time it takes for  $\mu$ Torrent to attain decent speeds, but that only lasts for the first few minutes at worst. After several minutes have passed, the connection limit will have already been reached if there are enough peers anyhow, so at that point, there is no difference between a low half-open connection limit and a high limit.

High half-open limits, on the other hand, cause connection problems for many users because Microsoft limits the number of half-open connections able to be made globally in certain versions of Windows (starting with Windows XP with SP2, up to Windows Vista with SP1). It can be patched, but for little gain, as already explained previously (and Microsoft resets the limit every so often with some Windows Update anyhow). Even if your half-open connection count isn't limited by TCPIP.sys (either through patching, or you're using an OS that doesn't impose such a limit), an increase in the half-open limit would fall into the

exact same limitations in efficacy as described above.

The rationale behind Microsoft's decision to limit half-open connections is that relatively few legitimate and properly-designed applications can be expected to require many half-open connections. On the flip side, if an application is attempting to establish a large amount of connections in a very short period of time, it is usually a sign that the application is a piece of malware attempting to communicate with other computers.

Why anyone would recommend that users increase their half-open limits as a general speed tweak that everyone should apply is incomprehensible. It doesn't actually help with speeds beyond the first few minutes at most, and comes with the detrimental effect of potentially killing connections for many users. The only potential exception to the "you don't need to increase your half-open connection limit" rule of thumb is if you have a connection with a very fast upload rate (think "several megabits per second"), and have many torrent jobs started simultaneously. In that situation, it may be beneficial to increase the half-open connection limit in order for  $\mu$ Torrent to be able to communicate (within a reasonable amount of time) with the large pool of peers and trackers associated with starting many torrent jobs simultaneously.

In instances where raising the half-open connection limit is unavoidable or necessary, the user should keep in mind that  $\mu$ Torrent's `net.max_halfopen` limit should never be set to a value greater than approximately 80% of the TCPIP.sys half-open connection limit. So if the TCPIP.sys limit is 100, the highest anyone should set  $\mu$ Torrent's `net.max_halfopen` should be 80. That doesn't mean it needs to be 80% either -- it can be less. That's just a safe limit guideline to maintain quality of connection.

## Why am I still receiving incoming connection attempts after I have already closed $\mu$ Torrent?

In BitTorrent, peers disconnect from the swarm without notifying

every other peer in the swarm because it would be inefficient to do so, and also because peers rarely have the IPs of every other peer in the swarm (which would make it impossible to do in the first place). Because most BitTorrent clients cache peer information, when they try to connect to a particular peer in their cache, they may end up attempting to connect to a peer that has already disconnected since they last received the peer's IP and port number.

This phenomenon is what you are observing when you see incoming connection attempts after you've exited  $\mu$ Torrent. Depending on how many peers had a copy of your IP and port number in their peer cache, this can take up to several weeks to subside. Having **DHT enabled** may magnify the effects, as there are many more peers using DHT than there are connected to any one particular torrent swarm (so more users may have your IP and port number by the time you disconnect). Although there is no way to prevent this from occurring, disabling DHT may reduce the amount of time needed for this to taper off. Assuming your network is properly protected, these connection attempts can be safely ignored.

[« Previous](#)   [Next »](#)

[µTorrent User Manual > FAQ: Frequently Asked Questions](#)

# Troubleshooting

## How should I get started in fixing my problems?

The first and most important action to take is to make sure you have everything in  $\mu$ Torrent configured properly as prescribed by [the setup guide](#) and any associated guides (like the [port forwarding](#) guide). As surprising as it may sound, many troubles that seem unrelated to one another stem from a misconfiguration issue. Additionally, make sure you are using the latest version of  $\mu$ Torrent available from [the download page](#), since bugs get fixed regularly from version to version (and the support staff does not provide support for older versions). If this initial step doesn't help, try looking through these troubleshooting FAQs to see if the problem has been addressed.

## Why are my torrent jobs missing?

First, make sure you aren't looking at [an improperly-configured list-view](#), and make sure you've selected an appropriate category in the [category list](#).

Assuming the torrent jobs list is truly empty, then the problem is caused by an improper shutdown of  $\mu$ Torrent, either due to a crash or a forced exit. The first thing you will need to do is reopen all of the `.torrent` files in the [settings directory](#) (or the [.torrent file storage location](#)). Afterwards, you should try enabling `bt.graceful_shutdown`. By turning this option on, you are allowing  $\mu$ Torrent the leisure of quitting cleanly, which generally solves the problem.

If  $\mu$ Torrent is crashing, and this behavior is a result of the crashes, then you should [try to solve the crashes](#).

## Why are my torrent jobs transferring so slowly?

## Can't I speed them up?

Make sure you've followed [the setup guide](#) carefully.

## Why can't $\mu$ Torrent connect to any DHT nodes?

The problem most likely occurs because something is blocking  $\mu$ Torrent from contacting other nodes. Try the following suggestions to see if they help:

- Make sure that when you [forwarded your port](#), you forwarded it for UDP connections in addition to TCP connections, since DHT makes heavy use of UDP.
- If you are using PeerGuardian (or an equivalent IP blocker), you might need to stop using it, or make an exception in the software for  $\mu$ Torrent's DHT bootstrap nodes at [router.utorrent.com](#) and [router.bittorrent.com](#), as  $\mu$ Torrent makes use of DHT nodes at those addresses to get the IP addresses of other nodes in the DHT network.
- Try adding a .torrent file from [Depthstrike.com's mirrors for open-source/freeware projects](#) to  $\mu$ Torrent's torrent jobs list. These .torrent files contain other DHT nodes that  $\mu$ Torrent can use to bootstrap onto the DHT network.
- Try removing `dht.dat` and `dht.dat.old` from the [settings directory](#), as these files might have been corrupted.

## Why can't I see anything in the directory browser dialog?

An incompatibility is causing the problem. To fix the issue, enable [gui.compat\\_diropen](#).

## Why do I get an "Invalid menu handle" error when I try opening a .torrent file from Firefox?

There may be some corruption in your Firefox preferences that causes this error to occur. In Firefox, check the "Tools" > "Options" > "Content" > "Manage" list for the TORRENT entry and remove it. Confirm the change in Firefox.

## Why do my torrent jobs grind to a halt with "Disk Overloaded" whenever I add a new one?

The problem occurs because of a design limitation in  $\mu$ Torrent that should be fixed in the future. There is nothing you can do except to wait for the problem to go away.

## Why does $\mu$ Torrent create or download parts of files I set to "Don't Download?"

This occurs because BitTorrent has no concept of files, only pieces. Because multiple files can share the same piece, and  $\mu$ Torrent has to download an entire piece to check its hash, it will effectively download data for another file (regardless of whether it was skipped). There is nothing you can do about  $\mu$ Torrent downloading data for another file if the data is a part of a piece that also belongs to another file that you do want downloaded. What you can do is prevent the entire skipped file from being allocated by enabling [diskio.use\\_partfile](#).

## Why does $\mu$ Torrent get stuck at a certain percentage for a torrent job?

Check the [availability of the torrent job](#), the [number of seeds](#) there are in the swarm, the [the number of peers](#) there are in the swarm, and the [amount of wasted data](#) that you've downloaded.

- If the availability is below 1.0 and the number of seeds is high, then the torrent swarm is most likely fake, and was created by anti-P2P organizations.
- If the availability is below 1.0 and the number of seeds is low as well, then you will have to be patient, as the torrent contents may simply not be well distributed.
- If you are seeing a large amount of wasted data, check [this FAQ](#).

## Why does µTorrent keep downloading or uploading while a torrent job is paused?

Pausing a torrent job might not stop all traffic for it immediately because µTorrent attempts to finish receiving and sending any [queued pieces](#) first. Because pausing a torrent job keeps connections open, occasional bits of peer communication [may show up in the speed calculations](#).

## Why does µTorrent not ask me where to download files, or which files I want to download?

If you would like µTorrent to let you select files to download before it actually adds the torrent job to the list, make sure you enable [the Add New Torrent dialog](#).

If µTorrent adds files without asking you anything, then that's because you [set a default download location](#), and so µTorrent assumes you would like to download everything to that directory. If you want µTorrent to always show the dialog when you manually open a .torrent file, then make µTorrent [always show the dialog on manual add](#).

## Why does µTorrent show less DHT nodes in the

## status bar than BitComet or Vuze?

µTorrent counts only the number of DHT nodes you are directly connected to. BitComet counts nodes that are one hop away from you (connected to nodes you're connected to), which inflates numbers. Vuze attempts to estimate the complete size of its DHT network.

## Why does µTorrent still calculate an ETA when it is seeding?

When a torrent job is in seeding mode, [the ETA column](#) estimates the amount of time it will take for µTorrent to reach [the seeding goal](#).

## Why does µTorrent still download a little bit when it is seeding, or even when no torrent jobs are started?

This behavior is absolutely normal, and occurs because µTorrent includes the communication overhead when calculating speeds. When you are uploading data to a peer, you continue to download a little bit of information from the peer to keep track of the peer's progress. Additionally, if the peer requests information or data from you, you also have to download that request (which gets figured into the download speed).

While it is normal to see a non-zero download rate while µTorrent is seeding, the behavior can be aggravated by improper settings. When you have too many connections established simultaneously, µTorrent has to communicate with more peers, and so there will be more downloading of requests and such. To optimize the situation, make sure you've selected the proper setting in [the Setup Guide](#).

Another possible cause for this behavior is the use of [the DHT network](#). Because DHT is always active as long as it is enabled, it will continue to download a little bit of data to pass around as a node

participating in the DHT network. This continues to occur even if no torrent jobs are started, or even if none of the torrent jobs in the list use DHT.

## Why does my Internet connection slow down or stop working while µTorrent is running?

Make sure you've checked your computer for [incompatible software](#) known to cause Internet disconnection. Additionally, check if your router is a ["bad" router](#). If you don't have any incompatible software installed, there are some suggestions you may want to try. Note that between any of the suggestions, you should restart your computer and check if µTorrent continues to cause Internet disconnections.

- Make sure you've followed [the setup guide](#) carefully
- Try disabling [DHT](#)
- Try disabling [UPnP](#) and [NAT-PMP](#)
- Try disabling [IP resolving](#)
- Try disabling [peer.resolve\\_country](#)
- Try lowering [net.max\\_halfopen](#) to 4 or 2
- Try lowering [bt.connect\\_speed](#) to 10 or 5
- Try lowering [the global maximum number of connections](#) to 100 or 50
- Try setting [bt.transp\\_disposition](#) to 5
- Try [patching your TCPIP.sys file](#) to a value greater than your [net.max\\_halfopen](#) limit (only if you are using Windows XP with SP2 or SP3, Windows 2003 with SP1 or SP2)

- Try connecting your computer directly to see if the problem persists

## Why doesn't $\mu$ Torrent obey the rate limits I selected?

Try enabling [Limit local peer bandwidth](#).

If you have limited your download rate, but  $\mu$ Torrent isn't obeying it, then take note the fact that  $\mu$ Torrent cannot control how fast peers send data to it. As a result, download rate limiting is often inaccurate. In addition, setting download rate limits may cause your upload rates to suffer because of the way download rate limiting had to be implemented.

## Why doesn't $\mu$ Torrent open .torrent or .btsearch files even though it is associated with them?

If  $\mu$ Torrent doesn't automatically open .torrent or .btsearch files when you [double-click](#) them, try pressing the relevant button(s) in [the Windows Integration preferences](#). If that doesn't help, you may have to remove the .torrent file type from Windows. In Windows Explorer, check the "Folder Options" > "File Types" list for the TORRENT entry and delete it, then press the associate button again from the preferences.

## Why doesn't $\mu$ Torrent open my web browser wherever it is needed?

This problem is generally caused by a misconfiguration in Windows with regards to which browser is default. Visit "Start" > "Control Panel" > "Add or Remove Programs" > "Set Program Access and Defaults" > "Custom." Where it asks you to select a default web browser, make sure you have your preferred browser selected, and if possible, make sure "Enable access to this program" is checked as well.

## Why doesn't µTorrent report me as a seeder when selectively downloading?

By definition, a seeder is a peer with all of the files fully completed. If you are missing any bit of data (which would occur if you skip any file), it would be incorrect for µTorrent to report you as a seeder. You are not a seeder unless you have 100% of the data.

## Why don't the move up or down buttons not move the torrent jobs?

These buttons change the queue order for the selected torrent jobs, not the actual order in which they appear in the list. If you would like these buttons to visually change the torrent job's order in the list, then you should sort the list by the # column.

## Why is there a large amount of wasted data being downloaded?

In many cases, this is an indication that the swarm you are connected to is fake or poisoned and set up by some anti-P2P organization. You might want to consider finding another source for the data you are trying to download. If this occurs with many torrent jobs, the problem could be an indication that your hardware is bad. Bad RAM, hard drive, or hard drive cables have been known to cause hashfails in µTorrent. Another source of hashfails may be your router. Some routers (like D-Link's) have been known to corrupt data when the router has DMZ (game) mode enabled, thus preventing users from completing downloads.

## Why is µTorrent crashing?

Make sure you've checked your computer for [incompatible software](#) known to cause crashes in µTorrent.

## Why is µTorrent now downloading when I was previously seeding?

This occurs if any of the files you're uploading are modified between the time you began seeding and the next time µTorrent re-checks the data. If a file changes, it causes the piece(s) containing it to fail the hash check, and accordingly, µTorrent will throw the piece out (thus causing the switch to downloading mode). Make sure you do not edit any files you are seeding. If you find yourself forgetting this, then consider setting the files you download to read-only mode once you are in seeding mode. If you did not manually edit any files, then there are some common causes for this problem:

- **thumbs.db:** On some systems, Windows automatically creates and updates a database of thumbnails for the media files in each folder called `thumbs.db`. If a `thumbs.db` file is a part of the torrent contents, any updates to it will cause hashfails in µTorrent. In Windows Explorer, check the "*Folder Options*" > "*View*" > "*Do not cache thumbnails*" option to prevent this from happening.
- **Media file tags:** Some media players are known to automatically edit tags on media files they play without user intervention. This may cause pieces to change (and consequently, hashfails to occur in µTorrent).

## Why is µTorrent using so much CPU?

Make sure you've checked your computer for **incompatible software** known to cause high CPU usage in µTorrent.

If you don't have any incompatible software installed, get **Process Explorer** and run it. If you see the DPCs are using significant amounts of CPU, then you may have some other buggy software installed, or perhaps buggy drivers and/or hardware. You can try running **RATTV3** to find the source of the DPCs. The RATTV3 output can be found in

%SystemRoot%\system32\LogFiles\RATTV3.

Another possible source for high CPU usage may be the transfer mode that your hard drive controller may be using for your hard drives. Check the "System" applet in the Windows Control Panel for the Device Manager. In the Device Manager, look for the IDE ATA/ATAPI controllers and check the properties for each of your primary and secondary IDE channels to see if any of their current transfer modes are in PIO. If so, this may be the cause of your problems. To fix it, try uninstalling the affected IDE channel and restarting your computer.

## Why is µTorrent using so much memory?

Make sure you've checked your computer for [incompatible software](#) known to cause high memory usage in µTorrent.

If you don't have any incompatible software installed, try tweaking [the disk cache settings](#). Of special note, if you are transferring data very quickly, try disabling the Windows cache for [disk writes](#) and [disk reads](#).

## Why is my firewall reporting connections being made by µTorrent on a port besides the one I chose?

Only incoming connections use the port you specify in the preferences. Outgoing connections use a random local port (called an ephemeral port); this is simply how TCP/IP functions, and is not a bug.

## Why is my firewall reporting that µTorrent is attempting to send e-mails or access the web?

In almost every case, this is a false positive generated by your firewall. Occasionally, peers use common service ports like 25 (SMTP), 80 (HTTP), or 110 (POP3) in order to bypass restrictions their ISPs may

impose on them. Because your firewall incorrectly assumes that any traffic with a destination port being a service port is traffic of the corresponding service, it (improperly) flags  $\mu$ Torrent's traffic accordingly. The one exception to this is when  $\mu$ Torrent checks for updates, in which case it really is attempting to access the web.

Assuming your computer itself is clean of malware, and you have obtained  $\mu$ Torrent from a known legitimate source (like [\\(\mu](#)Torrent's own download page), then it is safe to ignore these warnings and allow  $\mu$ Torrent to perform the action. If you do not want  $\mu$ Torrent to access these ports, you can set `bt.no_connect_to_services` and `bt.no_connect_to_services_list` accordingly.

## Why is the list-view I'm looking at blank?

There should still be a bar at the top of the list-view. `Right-click` the bar and select "Reset" in the context menu, and the apparently-missing details should be visible again.

## Why is there a .dat file in my download folder when the torrent contents don't contain such files?

This occurs when you perform selective file downloading and have `diskio.use_partfile` enabled. You can safely ignore the file, as it will be deleted after the torrent job is removed from the torrent jobs list.

## I've tried all kinds of suggestions, but still haven't been able to solve my problem. How should I proceed?

Feel free to stop by the [forums](#) or [IRC channel](#) to ask for help with your problems. When you ask for help, make sure you have the following pieces of information on hand and ready to be provided upon request:

- Everything you've tried so far in attempting to fix your problem
- The ISP you're using (preferably in addition to the type of connection you're using, like DSL, cable, satellite, etc.)
- The color of your [network status light](#)
- What [the Setup Guide](#) shows your current settings to be, along with the results of the port checker and speed test
- What you've set [net.max\\_halfopen](#) to, and what your TCPIP.sys half-open connection limit is set to
- The operating system you're using
- A log from [HijackThis](#) and/or [Process Explorer](#)
- The exact router and modem models you're using
- Some basic computer hardware specifications (motherboard and/or chipset, CPU, RAM, drive type, etc.)

[« Previous](#)   [Next »](#)

[μTorrent User Manual > FAQ: Frequently Asked Questions](#)

## Error Messages

### In a popup dialog, I get "Error: The device is not ready"

This error occurs if a drive  $\mu$ Torrent is trying to access does not exist, or is missing. This most frequently occurs if a drive letter has changed, or if the data  $\mu$ Torrent was looking for was located in a removable drive that is no longer in the computer. To resolve the issue, make sure the paths  $\mu$ Torrent is trying to use are all existent. This means that all torrent jobs should not have their **Save As** field set to a directory on some non-existent drive letter. The same thing applies to the **Directories** preferences.

If a torrent job is using a drive letter that no longer exists, make sure you **set the download location** for the job.

### In the Logger tab, I get "Error opening Windows Firewall"

This means that  $\mu$ Torrent was unable to add itself as an exception to the Windows Firewall using the standard Windows Firewall API. This occurs when Windows Firewall is disabled or not present, or that you are using an operating system that does not support the API (operating systems older than Windows XP SP2). The error is non-fatal, and can be safely ignored. If you want  $\mu$ Torrent to stop trying to add itself to the firewall exceptions list every time it starts, then you can disable the **Add  $\mu$ Torrent to Windows Firewall exceptions** option in the Preferences. If another firewall is present, you will still need to configure it properly to allow  $\mu$ Torrent to access the Internet.

### In the Logger tab, I get "NAT-PMP: Unable to map

## port with NAT-PMP"

This means that  $\mu$ Torrent was unable to map (forward) the port with NAT-PMP. If you have a **green status light** in the status bar, or the Setup Guide's port checker verifies that the port is open, then you can safely ignore this error. If otherwise, you'll need to manually **forward the port in your router**. If you want  $\mu$ Torrent to stop trying to map its listening port via NAT-PMP every time it starts, then you can disable the **Enable NAT-PMP port mapping** option in the Preferences.

## In the Logger tab, I get "UPnP: Unable to map UPnP port"

This means that  $\mu$ Torrent was unable to map (forward) the port with UPnP. If you have a **green status light** in the status bar, or the Setup Guide's port checker verifies that the port is open, then you can safely ignore this error. If otherwise, you'll need to manually **forward the port in your router**. If you want  $\mu$ Torrent to stop trying to map its listening port via UPnP every time it starts, then you can disable the **Enable UPnP port mapping** option in the Preferences.

## In the status bar, I get "Disk Overloaded"

This means that the disk was not able to keep up with the read/write speeds. To fix this, you should try tweaking your **Disk Cache** settings in the Preferences. Note that disk overloads may also occur if the device you are writing to or reading from is inherently slow by design (USB hard drives, for example). In such cases, tweaking the cache settings may help, but it isn't guaranteed to work.

If you get this error message when you add a new torrent job, consider it normal due to a design limitation in  $\mu$ Torrent (which should be fixed in the future). You can ignore the message, as it does not actually indicate any real disk overload. The message will disappear after a short amount of time.

## In the torrent job status, I get "Error:"

This means that an error occurred the last time  $\mu$ Torrent was run. Since  $\mu$ Torrent does not remember error messages, no error is specified. Try resuming the torrent job, or **forcing a re-check** to see if any error occurs again.

## In the torrent job status, I get "Error: Access Denied" and $\mu$ Torrent halts the torrent job

Make sure you are not using **a misconfigured incompatible software** that can cause file access issues. If you are not, then make sure you have the right credentials to be using the target file and/or (containing) directory.

On some systems, Windows Explorer may attempt to preview media files when you are browsing the containing directory for the files. If this happens while  $\mu$ Torrent is attempting to write to the file, then  $\mu$ Torrent won't be able to access the file, and accordingly, it will throw this error message. To prevent this from happening, make sure you don't view the folder in "Thumbnail" or "Filmstrip" mode. Additionally, you may want to consider unregistering the shell media file property extractor in Windows by visiting "Start" > "Run" (or by pressing **Windows+R** on the keyboard), typing `regsvr32 /u shmedia.dll` into the "Open" field, and pressing "OK."

## In the torrent job status, I get "Error: Data Error (cyclic redundancy check)" and $\mu$ Torrent halts the torrent job

This error is not an error generated by  $\mu$ Torrent, but rather, an error it receives from the hard drive. The error indicates that the hard drive was unable to read or write the data because the sectors are faulty, or are about to die. Cyclic redundancy check (CRC) errors are often signs of possible future disk corruption. Running `chkdsk` might help fix the

problem, but stronger recovery tools (such as SpinRite) may be required (assuming the disk is at all repairable).

## **In the torrent job status, I get "Error: Element not found" and µTorrent halts the torrent job**

This error may occur when you remove or rename files from a torrent job. If this is the case, try to re-add or rename the files back, or [force a re-check](#) on the torrent job.

## **In the torrent job status, I get "Error: The requested operation could not be completed due to a file system limitation" and µTorrent halts the torrent job**

This error occurs because of a problem with the way Windows Vista handles sparse files and NTFS compressed. When a sparse or compressed file reaches this (currently unspecified) limitation, all writes to the file will fail. The only way around this is to make a copy of the file, delete the existing (sparse/compressed) file, and replace the deleted file with the copy. It is recommended that you disable [diskio.sparse\\_files](#) if you are running Windows Vista. If you would like µTorrent to not fully allocate a file upon write while using Windows Vista, use [bt.compact\\_allocation](#) instead.

## **In the torrent job status, I get "Error: Not enough free space on disk" when I have more than enough free space**

This error occurs only on drives or partitions formatted to FAT32 because files greater than 4 GiB cannot be created on FAT32 partitions. The only fix is to convert the drive or partition to NTFS, or use another drive or partition that is already using NTFS.

To convert a drive to NTFS, read Microsoft's knowledge base article [KB307881](#).

## **In the torrent job status, I get "Error: Parameter is incorrect" when selectively downloading**

The cause of the error message is currently unknown, though it is being looked into. There is no workaround apart from not using selective file downloading on these operating systems if you receive the error. The error should not happen again if you simply resume the torrent job.

## **In the torrent job status, I get "Error: The process cannot access the file because it is being used by another process" and µTorrent halts the torrent job**

See the question regarding [Error: Access Denied](#).

## **In the torrent job status, I get "Error: The requested operation cannot be performed on a file with a user-mapped section open" and µTorrent halts the torrent job**

Make sure you are not using [a misconfigured incompatible software](#) that can cause file access issues.

## **In the torrent job status, I get "Error: The system cannot find the path specified" and µTorrent halts the torrent job**

This problem may be caused by a limitation in Windows with path lengths. Windows limits the maximum path length (including filenames)

to 255 characters. Try saving the torrent contents to a location closer to the drive's root, such as C:\Downloads\.

## **In the torrent job status, I get "Error: Unable to save the resume file"**

See the question regarding [Error: Access Denied](#). If that isn't relevant or doesn't help, make sure the directory  $\mu$ Torrent is using as its [settings directory](#) exists.

## **In the tracker status, I get "A socket operation encountered a dead network"**

Check to make sure you don't have [an incompatible software](#) installed. Previous reports on this error message indicate that [BitDefender Firewall](#) can cause such issues. There is no known way to fix the issue other than to uninstall the firewall (and replace it with another one if necessary).

If you are not using any software firewall, make sure you have the latest drivers installed for your network card, since this problem may well be caused by buggy drivers. It may also occur if your network gets physically disconnected (like if the network cable is disconnected, or if your modem or router shuts off).

## **In the tracker status, I get "An operation on a socket could not be performed because the system lacked sufficient buffer space or because a queue was full" and $\mu$ Torrent halts**

Check to make sure you don't have [an incompatible software](#) installed. Previous reports on this error message indicate that [Norton GoBack](#) can cause such issues. Updating the application or uninstalling

it entirely should solve the problem.

There is a Windows Registry entry in Windows 2000/XP/2003 that can cause this error. Read Microsoft's knowledge base article [KB196271](#) for further details.

This error can also be a symptom of improper configuration. Check [this FAQ entry](#) regarding interrupted connections, as the suggestions may solve this problem.

---

## **In the tracker status, I get "Connection closed by peer"**

In general, this error has manifested itself in cases where the user's ISP is interfering with BitTorrent tracker communication. Consult the [Bad ISPs](#) list on AzureusWiki to check if your ISP is known to interfere with BitTorrent traffic.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > FAQ: Frequently Asked Questions](#)

# Incompatibilities

What programs have been known to cause problems with µTorrent?

PROGRAM	KNOWN PROBLEM(S)
Anti-malware	
avast!	crashes or freezes, high CPU usage
ESET NOD32 Antivirus	crashes or freezes, Internet disconnection, tracker connectivity issue
McAfee VirusScan	missing interface text
Norman Antivirus	crashes or freezes
Norton AntiVirus	missing interface text
PC Tools Spyware Doctor	crashes or freezes, high CPU usage
PC Tools ThreatFire	file access issues
VCOM System Suite	crashes or freezes
Firewall	
BitDefender Firewall	crashes or freezes, tracker connectivity issue
FRITZ!DSL Protect	crashes or freezes
iolo System Mechanic Personal Firewall	crashes or freezes, unresponsiveness
Kerio Personal Firewall	high CPU usage, Internet disconnection
McAfee Personal Firewall	crashes or freezes, high CPU usage
NetPeeker	Internet disconnection
Norman Personal Firewall	crashes or freezes, high CPU usage
NVIDIA Firewall	crashes or freezes, high memory usage
Sunbelt Personal Firewall	high CPU usage, Internet disconnection

Indexing	
Google Desktop Search	file access issues
Nero Scout	file access issues
Roxio Media Indexer	file access issues
Windows Desktop Search	file access issues
WinZip QuickFind	file access issues
Miscellaneous	
CYBERSitter	high CPU usage
Microsoft Firewall Client for ISA Server	unresponsiveness
MouseImp	crashes or freezes
Norton GoBack	tracker connectivity issue
SpamPal	high CPU usage
Ultra Network Sniffer (Analyzer)	unresponsiveness
VCOM Fix-It Utilities	crashes or freezes

## I have an incompatible software installed, and am getting disconnected from the Internet

- If you are using **ESET NOD32 Antivirus**, see [this bit of information](#).
- If you are using **Kerio Personal Firewall** or **Sunbelt Personal Firewall**, then the problem may be another manifestation of [the high CPU usage problem](#).
- If you are using **NetPeeker**, make sure you upgrade to a newer version, as a bug was fixed in NetPeeker v2.72 that would cause an accumulation of connections stuck in FIN\_WAIT\_2 state, eventually stopping all Internet connection.

## I have an incompatible software installed, and µTorrent becomes unresponsive

If you are using any software known to cause unresponsiveness, then try updating it to the latest version. If that doesn't help, then currently, the only option is for you to uninstall that application (and maybe use a compatible alternative) if you wish to use µTorrent. You may also want to try searching [the forums](#) for any workarounds that may not have yet been added here.

## I have an incompatible software installed, and µTorrent crashes or freezes

- If you are using **ESET NOD32 Antivirus** v2.7.x, you should either disable NOD32's IMON component, or add µTorrent to the exclusions list in the IMON setup's "*Miscellaneous*" tab. If you are using ESET NOD32 Antivirus v3.x, you can add µTorrent to the HTTP web browsers exclusions list in the "*Web Access Protect*" section of the antivirus and antispyware setup.
- If you are using **Norman Antivirus**, you should disable the Internet Protection component.
- If you are using **NVIDIA Firewall**, and you experience crashing on a multi-core CPU, then you should try upgrading your NVIDIA nForce drivers to a version that includes v73.12 or newer of the Network Management Tools. If that is impossible, then you must either uninstall NVIDIA Firewall (ForceWare Network Access Manager in the Windows Add or Remove Programs applet), or you must set µTorrent's processor affinity to a single core from the Task Manager's processes list context menu (make sure only one core is checked).
- If you are using any other software known to cause crashes, then try updating it to the latest version. If that doesn't help, then currently, the only option is for you to uninstall that application (and maybe

use a compatible alternative) if you wish to use  $\mu$ Torrent. You may also want to try searching [the forums](#) for any workarounds that may not have yet been added here.

## I have an incompatible software installed, and $\mu$ Torrent has missing text in various places in the interface

**Norton AntiVirus** and **McAfee VirusScan** each have a buffer overrun protection feature that causes this to happen. To fix the problem, disable the feature in the application.

## I have an incompatible software installed, and $\mu$ Torrent has trouble accessing files

- If you are using **PC Tools ThreatFire**, then the problem may occur because of the software's proactively defensive nature against zero-day threats based on application behavior. To prevent any problems, add  $\mu$ Torrent to ThreatFire's trusted applications list:
  1. Open the ThreatFire GUI
  2. Select "*Advanced Tools*" on the sidebar
  3. Click "*Custom Rule Settings...*" in the "*Advanced Rule Settings*" tab
  4. Add a new item to the "*Trusted Process*" list in the "*Process Lists*" tab
  5. Select  $\mu$ Torrent from the list, or manually find it in the lower section of the dialog
  6. Make sure the newly-added  $\mu$ Torrent process in the list is checked
  7. Press "OK" in the dialog
- If you are using any desktop search or indexing software (whether or not it is known to cause file access issues), then you should either configure the application's indexing service to exclude any directory  $\mu$ Torrent is using from being indexed. Such directories include any directory  $\mu$ Torrent is downloading to or uploading from, and  $\mu$ Torrent's [settings directory](#). If excluding the directories doesn't help, then currently, the only option is for you to uninstall that

application (and maybe use a compatible alternative) if you wish to use  $\mu$ Torrent. You may also want to try searching [the forums](#) for any workarounds that may not have yet been added here.

- Although indexing services are the most common source of file access issues in  $\mu$ Torrent, other real-time scanning softwares (like anti-malware packages) have also been known to sometimes cause access problems for users.

## I have an incompatible software installed, and $\mu$ Torrent has trouble connecting to trackers

- If you are using **ESET NOD32 Antivirus**, see [this bit of information](#). Tracker connectivity issues may manifest themselves in the form of a HTTP 400 error.
- If you are using **BitDefender Firewall**, then the only solution is for you to uninstall the software and find an alternative security software.
- If you are using **Norton GoBack**, try updating the software, as the issue is supposed to have been fixed in Norton GoBack v4.1. If it doesn't help, try exiting the system tray icon for Norton GoBack before you run  $\mu$ Torrent, as it has been known to cause problems in older versions of Norton GoBack.
- If you have are using any other software firewall, try uninstalling (not disabling) it. If the problem is fixed, please inform us on [the forums](#) about your discovery.

## I have an incompatible software installed, and $\mu$ Torrent uses a lot of CPU

- If you are using **avast!**, then the problem may manifest itself because of the software's P2P shield, which scans all P2P activity.

To disable the feature, perform the following:

1. Open the "On-Access Scanner" for avast!
2. Press the "Details" button for more options
3. Select the P2P shield on the window to the left of the "On-Access Scanner" window
4. Press the "Terminate" button
5. Confirm the changes

Alternatively, avast! has a "Customize" option besides the sensitivity slider through which programs can be excluded from the P2P shield. Using this to exclude  $\mu$ Torrent's executable may solve the problem.

- If you are using **Kerio Personal Firewall** or **Sunbelt Personal Firewall**, then the problem may manifest itself because the firewall is constantly resolving IP addresses. Since BitTorrent clients tend to use a large amount of connections, the firewall has to spend a lot of resources trying to resolve that many IPs. To disable the IP resolving, perform the following:
  1. Stop all torrent jobs in  $\mu$ Torrent
  2. Open up the firewall GUI
  3. Select the "Overview" tab
  4. Look in the "Connections" section
  5.  in the window where all of the programs currently using a network connection are listed and uncheck the "Resolve Address" option
- If you are using **SpamPal**, then try switching to SpamPal v1.594, which is the last version reported to have been working fine with  $\mu$ Torrent. Otherwise, the only option left is to uninstall SpamPal, as it is known to cause the same issue with other applications (it isn't limited to  $\mu$ Torrent only).
- If you are using **Spyware Doctor**, then try switching to Spyware Doctor 4, which is the last version reported to have been working fine with  $\mu$ Torrent. Otherwise, the only option left is to uninstall Spyware Doctor.
- If you are using any other incompatible software, then currently, the only option is for you to uninstall that application (and maybe use a compatible alternative) if you wish to use  $\mu$ Torrent. You may also want to try searching [the forums](#) for any workarounds that may not

have yet been added here.

## I have an incompatible software installed, and $\mu$ Torrent uses a lot of memory

- If you are using **NVIDIA Firewall**, then you should try upgrading your NVIDIA nForce drivers to a version that includes v73.12 or newer of the Network Management Tools. If upgrading the firewall doesn't help, then you must uninstall NVIDIA Firewall (ForceWare Network Access Manager in the Windows Add or Remove Programs applet) if you wish to use  $\mu$ Torrent.

[« Previous](#)   [Next »](#)

[μTorrent User Manual > FAQ: Frequently Asked Questions](#)

## Miscellaneous

### How can $\mu$ Torrent be so fast and small?

$\mu$ Torrent is programmed in C++ using custom-coded libraries, with the GUI being written using the Win32 API. It is compressed with UPX to bring the size down by approximately 50% from its normal compiled size. A serious effort is made to keep the program as resource-efficient as possible.

### How do you pronounce $\mu$ Torrent?

Although there is no official way to pronounce  $\mu$ Torrent, Ludvig Strigeus wrote, "I usually say 'you torrent' because [the  $\mu$ ] looks like a u." He also offered "microtorrent" and "mytorrent" as alternative pronunciations.

### How do you write $\mu$ on the keyboard?

If you are using an international IME as your keyboard input language, press **AltGr** + **M**, or **Ctrl** + **Alt** + **M**. Alternatively, you can press **Alt** + **0181** (with the numbers pressed serially on the number pad), which works on all IMEs and regional settings.

### Is $\mu$ Torrent open source?

No, it is not open source, and it is very unlikely that it will ever become open source.

### Is there a Linux or Mac OS X version of $\mu$ Torrent?

You can download the Mac version of  $\mu$ Torrent [here](#). For those of

you wishing to run  $\mu$ Torrent on a non-Windows operating system, [the Wine project offers a solution](#).

## What is %AppData%?

%AppData% is a Windows environment variable that contains the path to the standard user application data directory. To access it, you can press "Start" button on the Windows taskbar, select "Run," type %AppData% into the "Open" field, and press "OK."

## Where can I get the latest $\mu$ Torrent beta?

Occasionally, public beta testing builds are posted on the download page, in the forum, or in the IRC channel. If there is no public posting... well, if you have to ask for the beta builds, then you can't get them :)

## Who makes $\mu$ Torrent?

- **Ludvig Strigeus** (ludde) is the original author and maintainer of  $\mu$ Torrent
- BitTorrent Inc.'s developers are the current maintainers of  $\mu$ Torrent. The developers include:
  - **Jan Brittonson** (CodeRed)
  - **Richard Choi** (rchoi)
  - **Adam Kelly** (AdamK)
  - **Arvid Norberg** (arvid)
  - **Ryan Norton** (RyanNorton)
- **Giancarlo Martínez** (Firon) maintains the  $\mu$ Torrent website and community
- **Timothy Su** (ignorantcow) is the website designer

## I have a bug report, feature request, or unanswered question. What should I do?

If you think you've found a bug, please make sure it is not caused by any [incompatible software](#). If the bug is reproducible, please visit the "Found Bugs" forum in the [µTorrent forums](#) and make sure your bug has not already been reported. If not, then register and post instructions on how to reproduce the problem.

If you have a feature you'd like to request, visit the forum and search to make sure it was not previously requested. If you post a request without first looking, then chances are fairly high that you will be told to search, and the thread will be locked.

If you have an unanswered question, visit the forum or IRC channel and ask. Searching the forum and reading the forum stickies are important things to do while at the forum.

[« Previous](#)   [Next »](#)

µTorrent User Manual

# Glossary

Especially if you're new to BitTorrent in general, there are probably many new and unfamiliar terms used throughout this manual (and in many other BitTorrent-related places). To help "clue" you in on the new vocabulary, this glossary can be of great help.

## A

### **announce**

The act of connecting to a tracker to update it on your status, and to obtain information from it as well, including (but not limited to) an updated **peer list**.

### **availability**

The number of complete copies of the torrent contents there are distributed in the part of the swarm you're connected to. The amount of the torrent contents you currently have is included in the availability count. A swarm with no **seed** and with an availability below 1.0 will likely be unable to finish transferring the complete torrent contents.

## B

### **block**

The units of data that comprise a **piece**. Because blocks do not directly affect whether torrent contents are considered to be finished transferring, it is not seen as an appreciable unit of data with regards to BitTorrent like the piece is.

### **byte**

A unit used for measuring the size of data on a computer storage device. Many people confuse "byte" for "bit" when referring to speeds. A byte is composed of 8 bits, so there is a

clear distinction, and terminology should not be confused when referring to bytes.

### **broadcatching**

The act of downloading content from an **RSS** feed.

## **C**

### **choked**

This word describes the state of a BitTorrent connection. When a connection is choked, it means the person who is supposed to be doing the uploading on the connection does not want to send anything. This generally happens when the uploader's upload slots are full.

### **client**

The application a user is using when connected to a **swarm**. In this case, the application being used to connect to swarms is  $\mu$ Torrent, so it is the client.

## **D**

### **DHCP (Dynamic Host Configuration Protocol)**

A **protocol** that allows networked devices to be assigned an unique **IP address** automatically from a pool of unused IP addresses.

### **DHT (Distributed Hash Table)**

A distributed tracker that works similarly to a regular **tracker** in that you **announce** to it and get back a list of peers that are transferring the same .torrent file as you. Because DHT is distributed, there is no single point of failure, so even if a single node disconnects from DHT, the tracker will continue to work (unlike with normal trackers, where if the server goes down, it becomes unusable). DHT can be thought of as a backup tracker.

**disk cache**

A feature that makes use of available memory to store data for quicker access as well as ease [disk thrashing](#). The use of a disk cache will cause an increase in memory usage in return for improved performance.

**disk thrashing**

When a storage disk gets accessed very frequently. Extended disk thrashing may lead to hard drive wear and tear, shortening a drive's life.

**double NAT**

A situation where the network device is behind more than one NAT devices (generally routers). In this situation, forwarding ports from just one of those NAT devices is generally insufficient, and more actions need to be taken as described in the [advanced port forwarding guide](#).

**download**

The act of transferring data from another computer onto your own.

**E****encryption**

The obfuscation (concealing) of data behind seemingly random data in order to hide its true identity.

**endgame mode**

A change in the piece requesting strategy that occurs when a download is near completion during which the client requests pieces from all connected peers rather than requesting a piece from one peer at a time in the normal operating mode. Endgame mode is used because download rates often slow down considerably as a torrent job nears completion due to the tendency for the remaining pieces to be downloaded from peers with saturated connections. By requesting data from all peers

rather than waiting for a single peer, such a bottleneck can be bypassed. This mode is not used during normal operating modes because of the large amount of overhead it potentially generates in sending requests to all peers.

### **ephemeral port range**

A range of port numbers automatically allocated by the operating system for use by any application on the system with network access. Ports in the ephemeral port range are typically used to make temporary outgoing connections. The default ephemeral port range is configurable via the Windows Registry, and may vary from (operating) system to (operating) system. More about the ephemeral port range can be learned on Wikipedia's [Ephemeral Ports](#) article and its external links.

## **F**

### **firewall**

A barrier (hardware and/or software) that prevents communication to and/or from certain computers, depending on the rules set in the firewall.

## **G**

### **GiB (gibibyte)**

A gibibyte is equal to 1024 [MiB](#). Most people are referring to "gibibyte" when they say "gigabyte," although that is technically an incorrect usage of terms.

## **H**

### **half-open connection**

A connection that is not fully established on both ends. Half-open connections occur when you attempt to connect to an IP

address, but the IP address hasn't yet responded.

### **hash**

A "fingerprint" of data assumed to be unique to the data. Because of the assumed uniqueness of the data, it is used to verify that a piece of data is indeed uncorrupted (since the corrupted data's hash would not match its expected hash).

### **hash check**

The comparing of a piece of data's hash with a reference hash in order to verify the integrity of the piece of data.

### **hashfail**

When a piece fails the hash check used to verify data integrity.

### **index**

A site that lists .torrent files available for download.

### **Initial Seeding (Super Seeding)**

A method of seeding that attempts to decrease the bandwidth load for the initial seeder. With normal seeding methods, the initial seeder typically has to upload 150% to 200%, or even more, of the original data in before a full copy of the data has been distributed into the swarm. With initial seeding, the initial seed attempts to get the rarest pieces out instead of uploading identical pieces repeatedly, often lowering the initial upload requirement to 105%. Initial seeding does not necessarily improve upload speeds or decrease seeding time. It should be used only if you are the sole seeder on the swarm, and if there are at least 2 peers connected. Generally, initial seeding should not be used by people with high upload speeds.

### **interested**

This word describes the state of a BitTorrent connection. When a peer is interested, it means the peer is interested in the data

that the peer on the other end of the connection has, and is willing to accept data from the other peer.

**IP address**

A number used to uniquely identify devices on a network.

**ISP (Internet Service Provider)**

The company providing for your Internet service.

**ISP throttling**

A term used to refer to the **throttling** of BitTorrent traffic by ISPs.

**K****KiB (kibibyte)**

A kibibyte is equal to 1024 **bytes**. Most people are referring to kibibyte when they say "kilobyte," although that is technically an incorrect usage of terms.

**L****LAN (Local Area Network)**

A network of computers in a local area, such as a home.

**LAN IP address**

The private, internal IP address that locates a computer on a **LAN**. A LAN IP address is not visible to users outside of the LAN. As described by RFC 1918, the following ranges are designated as reserved IP addresses for private LANs:

- 10.0.0.0 - 10.255.255.255
- 172.16.0.0 - 172.31.255.255
- 192.168.0.0 - 192.168.255.255

**LPD (Local Peer Discovery)**

A method by which µTorrent attempts to discover new peers local relative to your computer's network. Local Peer Discovery

makes use of IP multicast.

### **leecher**

A person who downloads, but fails to reciprocate the generosity of others by not sharing back. The word "leecher" carries a strong negative connotation. Some people use the words "leecher" and "peer" interchangeably, though this practice is not recommended (as it may lead to word confusion).

## **M**

### **magnet URI**

A link that tells the client what files to find and download over DHT.

### **MiB (mebibyte)**

A mebibyte is equal to 1024 KiB. Most people are referring to "mebibyte" when they say "megabyte," although that is technically an incorrect usage of terms.

### **Micro Transport Protocol (uTP)**

A UDP-based reliable transport protocol designed to minimize latency, but maximize bandwidth when latency is not excessive. This alleviates the bandwidth saturation that often occurs to BitTorrent users while they are transferring data and using the Internet for other purposes.

## **N**

### **NAT (Network Address Translation)**

The changing of the source or destination IP address for a data packet. This usually occurs when one is behind a firewall or router, where it translates IP addresses so that multiple computers can exist on a LAN with while using the same WAN IP address.

## **NAT-PMP (NAT Port Mapping Protocol)**

An alternative to UPnP created by Apple, Inc. NAT-PMP is not as widely supported as UPnP is, and uptake of the protocol has been limited to Apple, Inc. products only thus far.

## **NAT Traversal**

Techniques of establishing connections that traverse (pass through) NAT gateways. When it works, NAT traversal can help bypass port forwarding issues.

## **O**

### **optimistic unchoke**

When a client tries to start a transfer on a previously **choked** connection in hopes that the connection becomes unchoked.

### **overhead**

Additional data used and required for communication and coordination between sender and receiver that is not part of the **payload** data actually being transferred.

## **P**

### **P2P (peer-to-peer)**

The use of bandwidth of users using the same peer-to-peer service to perform the functions of the peer-to-peer service or software. Centralized servers are not what keeps P2P networks alive, but rather, the peers themselves.

### **payload**

The actual data being transferred from sender to receiver, not counting **overhead**.

### **PE (Protocol Encryption)**

An specification designed jointly by Azureus and µTorrent developers, created as an attempt to bypass throttling and/or

blocking of BitTorrent traffic by ISPs by encryption of the data. There are different methods of encryption, ranging from full encryption of all of the data, to partial encryption of the data (header encryption only, not unlike with PHE, although it's still not as easily detected as PHE).

### **peer**

A user/client connected to the swarm. People sometimes refer to peers as "leechers," though they also use the same word to refer to its more negative connotation. It's recommended that you use the word "leecher" to strictly refer to people who don't share so to keep the distinction clear and confusion to a minimum.

### **peer list**

A list containing the IPs and ports of other peers.

### **PEX (Peer Exchange)**

A feature to exchange peer lists with other peers that support the same PEX implementation (generally limited to peers using the same BitTorrent client). By exchanging peer lists, it's possible to find peers not included in the peer list supplied by the tracker.

### **PHE (Protocol Header Encryption)**

An old method of encryption created by the BitComet developer that encrypted only a part of the data (the header) in an attempt to bypass ISP throttling and/or blocking of BitTorrent traffic. Because its specification was designed in a relatively poor manner, ISPs were able to detect it with little trouble, rendering it useless.

### **piece**

The smallest appreciable unit of data in BitTorrent. The size of pieces can be different depending on the .torrent file in question.

### **piece distribution**

The general distribution of the pieces across the swarm. BitTorrent is generally most efficient when piece distribution is random, with minimal "clumping" of pieces available in the swarm.

### **poisoning**

The act of intentionally feeding invalid data into the swarm, resulting in **hashfails** for peers receiving the invalid data. Outfits with (or hired by other entities with) anti-P2P agendas are the most common sources of swarm poisoning.

### **port forwarding**

The act of passing data on the forwarded port from one network device to another. In most cases regarding BitTorrent, port forwarding refers to the forwarding of connections from a router to a specific computer attempting to listen on that port.

### **'private' flag**

A piece of information stored in a .torrent file that tells any BitTorrent client that recognizes the flag to disable **DHT**, **LPD**, and **PEX** for that specific .torrent. The 'private' flag is typically used in .torrent files served by **private trackers** as a method of keeping a swarm isolated from people who aren't members of the private tracker.

### **private tracker**

A **tracker** that requires users to log in to use it. Private trackers typically enforce ratio requirements (by banning users whose ratios are too low) in order to prevent or minimize the **leeching** that is prevalent on many **public trackers**.

### **protocol**

A set of rules and description of how to do things. In the case of the BitTorrent protocol, it is a set of rules describing how BitTorrent **clients** should communicate and transfer data with each other.

### **proxy**

A computer that is told to make a connection to another computer, and relay the data transferred between the two computers to the original computer that connected to the proxy. Essentially, using a proxy is a way to make an indirect connection to another computer by way of the proxy computer.

### **public tracker**

A **tracker** that is open for anyone to use (as opposed to **private trackers**, where only people who hold accounts can use the tracker).

## **R**

### **ratio**

A number derived from the division of two other numbers. In the context of BitTorrent, people are normally referring to **share ratio** when they speak of a ratio.

### **reseed**

The act of rejoining a swarm with no seeds as a **seed**.

### **RSS feed**

A file that is updated so that it delivers information and content in such a way that allows one to track updates quickly and easily.

## **S**

### **scrape**

The grabbing of statistics (number of seeds and peers) from a tracker regarding a specific swarm.

### **seed**

A **peer** with 100% of the data in the torrent contents.

### **seeding**

The act of being connected to a **swarm** as a **seed**.

### **share ratio**

The ratio of the amount of data you've uploaded to the amount of data you've downloaded.

### **snubbed**

This word describes the state of a BitTorrent connection. A connection is marked as snubbed when the client has not received anything over the connection for an extended period of time.

### **static IP address**

An IP address that does not change (remains static) across multiple sessions. A static IP address is necessary in **port forwarding**, as ports are usually forwarded to a specific IP address, where the rule does not change even if the computer's IP address does.

### **swarm**

The collective group of peers (which includes seeds) that are connected by a common .torrent file.

## **T**

### **throttle**

A term used to refer to the intentional slowing down of transfer rates (download and/or upload), typically used in the context of **ISP throttling**.

### **torrent**

A small file containing metadata from the files it is describing. In other contexts, it is sometimes used to refer to the **swarm** connected around that small file.

### **tracker**

Something that a **client** connects to in order to share its IP and

port, as well as obtain information, including **peer lists**.

## U

### **upload**

The act of transferring data from your computer onto another.

### **UPnP (Universal Plug and Play)**

A protocol that allows devices on a network to communicate with each other seamlessly. In the case of  $\mu$ Torrent, UPnP is used to **forward a port** on a router without the need to open the port manually.

## W

### **WAN (Wide Area Network)**

A computer network that covers a large geographical area. A WAN connects multiple **LANs** together. The Internet is an example of a WAN.

### **WAN IP address**

The public, external IP address that users outside of your own network see your network to be located at on the **WAN**. WAN IP addresses reveal nothing about internal IP address allocation on a **LAN** located at the WAN IP address.

### **wasted**

Data that is tossed out either because it **hashfailed**, or because it was redundant data that the client had already downloaded.

### **web interface (Web UI)**

An interface for a supported web browser that allows one to control an application remotely.

### **web seed**

A **seed** that is basically a regular web server hosting the

requested file. BitTorrent clients that support web seeds use them like any other seed, and can request data segments from the server much like requesting pieces from an ordinary seed. The use of web seeds ensures that a torrent swarm will never die as long as the file being seeded is left intact on the server and the server does not go down.

[« Previous](#)   [Next »](#)

µTorrent User Manual

# External Links

Here's a list of external webpages mentioned in the help manual:

- **µTorrent**

- Webpage
- Skins Page
- Download Page
- Forums
  - flags.conf/flags.bmp
- IRC Channel

- **VuzeWiki**

- Advanced Network Settings
- Bad routers
- Bad ISPs

- **Depthstrike.com Resources**

- Mirrors for Open-Source/Freeware Projects
- Multitracker Specifications

- **Microsoft**

- KB196271: When you try to connect from TCP ports greater than 5000 you receive the error 'WSAENOBUFFS (10055)'
- KB307881: How to convert a FAT16 volume or a FAT32 volume to an NTFS file system in Windows XP
- KB894564: How to change the binding order of network adapters in Windows XP and in Windows 2000
- RATTV3

- **Miscellaneous**

- Ephemeral Ports
- HijackThis
- Process Explorer
- TCP/IP.sys patcher (LvlLord's EventID 4226 Patcher)
- Why is Being Firewallled Bad?

- **PortForward**

- [How to Set Up a Static IP Address](#)
- [Router List for PortForwarding Guides](#)
- [Motorola SurfBoard Issues](#)

- **Running on Alternative Operating Systems**

- [Apple XCode](#)
- [Darwine](#)
- [WineHQ.org](#)

- **Speed Test Torrents**

- [µTorrent; Featured Torrents](#)
- [OpenOffice.org](#)
- [Slackware Linux](#)
- [Ubuntu Linux](#)

[« Previous](#)   [Next »](#)

µTorrent User Manual

# Changelog

Recent changes to the  $\mu$ Torrent help files:

## 3.3.2 Update, August 17, 2013

- Fixed typos/links

## 3.3.1 Update, June 11, 2013

- Updated  $\mu$ Torrent icons
- Added new metadata fields and updated Create New Torrent dialog
- Added description for Disk Timing graph
- Added description for auto-update options
- Added "Disk Job" name and description
- Added advanced feature: `diskio.quick_hash`

## 3.3 Update, May 1, 2013

- Added keyboard shortcut for torrent filter
- Added icon and description for torrent filter
- Removed reference to Apps

## 3.3 Update, March 20, 2013

- Removed references to BitTorrent DNA
- Removed references to Apps
- Added references and links to  $\mu$ Torrent for Mac
- Changed references to Azureus to Vuze
- Removed references to Find Content page
- Added  $\mu$ Torrent Featured Torrents to list of test torrents
- Cleaned up list of advanced parameters, added additional definitions and removed unused and undefined entries



« Previous

µTorrent User Manual

# Credits

This user manual was authored by **Ultima**, **Rafi** and **Firon**. That is all. Or not...

Yeah, it's taken *such a long time* to get this manual to where it currently is, but without help from a bunch of people, I'm sure it would've taken ages longer. Let this page be a "thank you" for helping! Each list is sorted in alphabetical order.

- **Developers** (*The reasons  $\mu$ Torrent exists*)
  - **Iudde** (Ludvig Strigeus)
  - **BitTorrent, Inc.**
- **Overall** (*Helped with a broad range of things*)
  - **DreadWingKnight** (Harold Feit): Being someone who always has definite answers :)
  - **Firon** (Giancarlo Martínez): Being someone who *usually* has definite answers xD
  - **ignorantcow** (Timothy Su): Layout :D
  - **Smoovious**: Tolerating and answering all of my stupid questions ;]
- **Guides** (*Played major roles in shaping the guides*)
  - **jeroenimo** and **NoOneButMe**: Darwine information!
  - **Pelo**: New and improved guides!
  - **splintax**: Original guides
  - **TheSHAD0W**: "[Why Is Being Firewalled Bad?](#)" article
- **Proofreaders** (*There'd be even more mistakes abound without 'em*)
  - **emc**: Lots of random suggestions... :)
  - **icleolion**: Anything else I should fix? xP
  - **moogly**: Funny it seems like it's mostly non-native English speakers proofreading and fixing my typos... :D
  - **schnurlos**: Caught too many minor mistakes! x[
- **Miscellaneous**
  - **Forum lurkers** ;)
  - **IRC idlers** ;D

If you think you weren't properly credited, feel free to bug me about it on IRC or the forums and I'll make the judgement call. If I do find that I've made a mistake, forgive me, and don't feel too bad about me failing to remember properly -- as I've said, it's been a long while since I started writing this manual, and I can be forgetful a lot of times :P

[« Previous](#)

[« Previous](#)   [Next »](#)

[μTorrent User Manual](#) > [Appendix A: The μTorrent Interface](#) >  
[Main Window](#) > [Detailed Info Pane](#)

# Related

 In the Related tab you can view Related text that others have added to their torrents. The options are:

- **Website** - any websites related to the current torrent.
- **RSS Feeds** - any RSS feeds related to the current torrent.
- **Similar Torrents** - any torrents related to the current torrent.

[« Previous](#)   [Next »](#)