

## Frequently Asked Questions about mIRC.

**Latest mIRC version: 4.1 April 27th, 1996**

**The mIRC FAQ - version 25 revision 0**

**Last FAQ Update: April 27th, 1996**

This FAQ attempts to answer the most frequently asked questions about the Winsock IRC client mIRC, developed by Khaled Mardam-Bey. This FAQ wont answer general IRC questions since lots of info is already available on that. I'm not the author of mIRC but I've spent quite some time beta testing it. Over the last few months, I've come to love mIRC for its small size and useful tools... in particular, the popup menus and the Remote section are really great! ...

This FAQ as well as mIRC is still a work in progress! Most subjects in this file were prompted by questions in the Usenet newsgroups alt.irc and alt.irc.questions. Please continue to post your questions there (and answers as well), as those newsgroups are my learning place as well. ;-) Also feel free to provide additional information or corrections for the FAQ. I don't have all the answers either!

The first part (Sections 1 - 6) of this file is the actual FAQ. The last part (Sections 7 + 8) consists of a tutorial or reference manual for mIRC's programming features. If you want to learn the 'what and how' of creating Aliases, Popups and Remote Commands and Events in mIRC, check out the last part of this file. I can highly recommend these sections to you all !

Thanks to all the people who voluntarily contributed to this FAQ. In particular Mookies and Bryan for making additional parts and html-ing. Shorty, Keyman and Michael, thanks for fixing a lot of spelling and grammar bugs!

You can e-mail me with FAQ related remarks at: [mirco@dds.nl](mailto:mirco@dds.nl)  
Questions about mIRC are best asked in the Usenet newsgroup alt.irc.questions  
As usual, direct all mIRC bug reports to Khaled Mardam-Bey at:  
[khaled@mardam.demon.co.uk](mailto:khaled@mardam.demon.co.uk)

If you have further questions about mIRC, please visit the IRC channel #mIRC on EFnet, Undernet or Dalnet. BUT, if you ask a question that is really well handled by this or other FAQ's, then please accept that you be pointed back to this or other files.

This FAQ is also available on the World Wide Web on the mIRC Homepages :

<a href="http://www.mirc.co.uk/">http://www.mirc.co.uk/</a>	(London, United Kingdom)
<a href="http://www-2.nijenrode.nl/ftp/pub/mirc/">http://www-2.nijenrode.nl/ftp/pub/mirc/</a>	(Nijenrode, The Netherlands)
<a href="http://www.geocities.com/Colosseum/1822/">http://www.geocities.com/Colosseum/1822/</a>	(Beverly Hills, California)
<a href="http://www.sbcomp.com/mirc/">http://www.sbcomp.com/mirc/</a>	(Kansas City, Missouri)
<a href="http://www.ozemail.com.au/~mooremt/">http://www.ozemail.com.au/~mooremt/</a>	(Sydney, Australia)
<a href="http://www.gil.com.au/~strike/">http://www.gil.com.au/~strike/</a>	(Ipswich, Australia)

**Copyrights** You are allowed to provide and distribute the mIRC FAQ -as is- by or on any medium as long as you make it available for free. You are not allowed to change anything in the file or charge any amount of money for your services. If you want to copy only certain parts for whatever use, make sure to mention my name and the FAQ as the source of information with every single quote whenever you publish it. Copyrights 1995, 1996 Tjerk Vonck [mirco@dds.nl](mailto:mirco@dds.nl)

## Table of Contents

- 1 [WHAT IS mIRC?](#)
- 2 [LATEST NEWS ON mIRC.](#)
- 3 [WHAT DO YOU NEED TO RUN mIRC AND WHERE TO GET IT.](#)
- 4 [SHORT INTRODUCTION TO IRC.](#)
- 5 [SOME SHORT NOTES AND TIPS ON mIRC.](#)
- 6 [FEATURES, TIPS and ANSWERS to QUESTIONS.](#)
  
- 6-1 [How do I run the 32-bit mIRC on win 3.1 or WFW311 ?](#)
- 6-2 [How do I run the 32-bit mIRC on Windows95 ?](#)
- 6-3 [How do I solve the "Can't Resolve Local Host Name" error ?](#)
- 6-4 [How do I solve the "Unable to resolve IRC server name" error ?](#)
- 6-5 [How do I solve the "Not enough user parameters" error?](#)
- 6-6 [How do I solve the "You haven't registered" error ?](#)
- 6-7 [Why am I unidentified and what does it matter ?](#)
- 6-8 [I can't get mIRC to DCC send or initiate a DCC chat !](#)
- 6-9 [I can't get mIRC to DCC get!](#)
- 6-10 [How do I Copy and Paste ?](#)
- 6-11 [How do I get colored text in mIRC ?](#)
- 6-12 [How can I customize the fonts used in mIRC ?](#)
- 6-13 [How can I see mIRC's 3-D dialog boxes ?](#)
- 6-14 [How do I specify people correctly in the auto-op and ignore list ?](#)
- 6-15 [How can I make mIRC react to Remote Commands ?](#)
- 6-16 [How can I make mIRC react to Events ?](#)
- 6-17 [How do I specify my friend Bill correctly in the User list with an access level of 2?](#)
- 6-18 [I get disconnected after the /LIST command.](#)
- 6-19 [Running multiple copies of mIRC.](#)
- 6-20 [How do I make myself invisible to other users ?](#)
- 6-21 [How does the new sound command in mIRC work ?](#)
- 6-22 [How does the WWW support in mIRC work ?](#)
- 6-23 [What is the built-in finger server for ?](#)
- 6-24 [How can I easily serve my collection of shareware to others?](#)
- 6-25 [How do I use mIRC's built in support for Text To Speech by Monologue ?](#)
  
- 7 [SOME NOTES ON 'PROGRAMMING' in mIRC.](#)
- 7-1 [The Aliases section.](#)
- 7-2 [The Popups section.](#)
- 7-3 [The Remote section.](#)
- 7-3-1 [Remote Users.](#)
- 7-3-2 [Remote Commands.](#)
- 7-3-3 [The Remote Events.](#)
- 7-3-4 [Various flags you can use in front of Command or Event lines.](#)
- 7-3-5 [Variables.](#)
  
- 8 [SOME SAMPLE MIRC SETTINGS \(aliases, popups, Remote, etc.\).](#)

## 1 What is mIRC?

mIRC is a freeware IRC Chat client for Windows. It is developed by and is copyrighted by Khaled Mardam-Bey. For those of you new to the Internet, IRC stands for Internet Relay Chat. The IRC network is a virtual meeting place where people from all over the world can meet

and talk (well, type). On IRC you meet others on "channels" (rooms, virtual places, usually with a certain topic of conversation) to talk in groups, or privately. There is no restriction to the number of people that can participate in a given discussion, or the number of channels that can be formed on IRC. As a user you run a "client" program like mIRC which connects to a "server" in an IRC network. All servers are interconnected and pass messages from user to user over the IRC network. One server can be connected to several other servers and up to hundreds of clients. If you need more information on IRC go to mIRC's www Homepages where a lot of additional information is given. mIRC attempts to provide a user-friendly interface for use with the IRC network. It has the following features:

- A powerful dynamic /help command.
- Netscape, Mosaic and MS Internet Explorer support .... surf the www waves !!
- An 'off you go' Toolbar.... and even Tooltips !
- A user programmable menu bar.
- A Remote CTCP Commands handler.
- A simple but smart Events handler.
- Colored text to ease reading.
- A handful of options.
- Uncluttered display.
- Simple, fully configurable aliases.
- Configurable popup menus.
- DCC Send/Get/Chat.
- Full configurable fonts.
- Built in Ident server.
- Finger client.
- A simple but powerful File server.
- Programmable Function keys.

The various parts of the program have been designed with the aim of simplifying and speeding up your IRC sessions. mIRC is made to be very configurable... there are a lot of simple switches you can use to personalize mIRC to your needs. And unlike a lot of other IRC programs mIRC still gets more and more mature every new version. You can define your own commands and implement your personal reactions to commands others give to your client. This all does not make mIRC a bot program, and it doesn't even support IRCii scripting and possibly a large number of other things... But it has much of the same functionality, thanks to the Remote Commands and Events and various other features... and what do you need a war script for, anyway?

I think mIRC will help you to focus on the main point ; IRC is for Chatting ....

[Table of contents](#)

## **2 Latest News on mIRC... (What is new in version 4.1 )**

mIRC's version 4.1 fixes the small (and larger) buggies that were found in the 4.0 version. Most of them just affected advanced users who were trying to make tricky remote implementations. The very obvious bug in the /sound command (oops) is gone, some .ini file write routines are improved, the DCC Get minimize option works properly now and several other thingies are updated/changed and fixed.

Some new and very convenient commands and options are added. Worth to be mentioned here are the new /links command and the new handling of .log files. Check them out ! Several little changes are made to improve command syntaxes and/or possibilities.

## Fixes, Changes and Improvements in version 4.1

1. The /sound command being limited to eight character channel/nick names is fixed.
2. Fixed the problem that the mirc.ini file was getting corrupted in a certain situation.
3. Fixed small servers sorting bug in the File/Setup/Servers dialog.
4. The /links command is improved a lot and now offers you fast acces to other servers.
5. Fixed a bug with the | character in nicknames and userids.
6. Changed /echo command, format is now: /echo [ColorNumber] <-s|WindowName> <text>
7. The DCC Get windows now correctly remembers its minimize option setting.
8. Fixed a bug that caused the \$ character in commands to be seen as a variable.
9. Window buffers are now saved to a .log file instead of to a .txt file. Also a log file maintnance box with view and delete options is added.
10. Fixed a problem with the "read" command in the fileserver.
11. The about dialog improved. (Disable it popping up under File/Setup/Options)

Read the mIRC help file to get more info on the new things mentioned here... also read the versions.txt included with all new mIRC versions for a complete list of fixes and improvements. To state them all here would make the faq tooo long.

## Fixes, Changes and Improvements in version 4.0

1. Nicks added to the notify list with /notify are now sorted into the list. In the notify section under File/Options you can now add little reminder notes to nicknames as well !
2. Users of Windows95 can properly access their right mouse button popup-menu in the edit box again.
3. The help menu has made dynamic to display all .hlp files in mIRC's directory, as well as the readme.txt, versions.txt, and update.txt files, for quick access. Also internal aliases are added that match the help filenames, so if you have the file "ircintro.hlp" in mIRC's directory you can type /ircintro <topic> ! (like /help <topic>) You can add whatever help files you like to mIRC's help menu! Both the mIRC FAQ and the IRC Intro file are available in windows help file format from the mIRC www pages.
4. To the File/Options/Event\_Beeps section the possibility is added to play a wav file instead of the usual, boring beep.
5. All /dde commands are now buffered before sending to other programs. All commands are subsequently sent to the clients at intervals specified by the DDE delay setting. This assures better and more trustworthy DDE performance.
6. A new mircy feature is found in the menu bar. A configurable menu item (!) is at your service which allows you to add your personal menu items to the main menu bar. You can compare it to mIRC's programmable popup menu's. The menu can have the same functionality as the main, query, and channel-names-list popup menus, and allows you to define commands that work in all different types of windows. This user programmable menu bar is maintained from the Tools/Popups/Menu\_Bar section.
7. Fixed sneaky ways little fingers found to get around the channel list hide and lock option.
8. The server setup dialog is improved a lot, as well as mIRC's behaviour during connection to IRC servers. You can define groups of servers which will be displayed together in the list of servers. Establishing a connection will be retried automatically now for a set number of times. You can select multiple servers to try to connect to, which will subsequently be tried. Using "/server groupname" which will cycle through all servers of group "groupname" when mIRC is trying to connect.
9. You can now auto-unignore an ignored person after N seconds. A new switch is built in into the ignore command. You can do "/ignore -auN <nickname> [type]",

which will first ignore (a=add) the person and then unignore the specified address after N seconds. You can also specify -what- to ignore exactly. With "/ignore -apcnti <nickname>", where p=privates, c=channel text, n=notices, t=ctcps, i=invites. The /ignore now \*adds\* an address by default, to remove an address you can use the -r (remove) switch. The ignore command does not toggle anymore!

10. mIRC now allows you to define all kinds of variables. Variables always start with a % and can have names of any length. The variables are stored between sessions in a variable file in the mIRC directory. Variables can contain numbers, words or even lines of text. On variables containing numbers you can apply mathematical manipulations to increase, decrease, add and subtract variables with given values or each other. Variables can be used in all remote commands, events, aliases and popups. You can use them to create download statistics, voting machines and gambling games ...use your imagination !!

11. The /sound command now doesn't require you to specify a message anymore. You can now send a sound request to a channel or person without an action text attached.

12. Support for SOCKS compliant Firewall's is added. You still cant use DCC Send, Get and Chat's through a firewall, but at least you can use mIRC from your company's network now!

[Table of contents](#)

### **3 What do I need to run mIRC ? Where do I get it ?**

Besides running MS-Windows you need to have an Internet account and a properly installed Winsock. If you can use FTP, E-mail, News or other Internet programs from within Windows already, you can safely assume you have a both. If you do not have a properly working winsock installed on your PC you should install one first. Information on Winsocks can be retrieved from your provider or found in The Windows Internet Guide by Michael Jason Mezaros at <http://twig.hypercon.com/master.html> or in Win95.com on <http://www.windows95.com>

Although mIRC is not spread by advanced distribution schemes of primary and mirror FTP sites, it is available from hundreds of places all over the world. Some of the major ones are :

UK: <ftp://ftp.demon.co.uk/pub/ibmpc/win3/winsock/apps/mirc/>  
USA <ftp://oak.oakland.edu/pub/irc/>  
USA <ftp://cs-ftp.bu.edu/irc/clients/pc/windows/>  
FI <ftp://ftp.funet.fi/pub/msdos/networks/irc/windows/>  
USA <ftp://papa.indstate.edu/winsock-l/winirc/>  
USA <ftp://ftp.undernet.org/pub/irc/clients/windows/>  
USA <ftp://ftp.winsite.com/pub/pc/win3/winsock>  
USA <ftp://www.windows95.org/pub/win95/mirc/>

The most up-to-date list of places where you can get mIRC is the Download page on the mIRC Homepages... You could also join the IRC channel #mIRC to get the latest version or to ask all remaining questions... (But: be very sure this FAQ doesn't answer your question)

[Table of contents](#)

### **4 Short Introduction to IRC.**

(Read more in the IRC Intro file available on the mIRC www pages)

**What is IRC** - IRC stands for "Internet Relay Chat". It was originally written by Jarkko Oikarinen in 1988. Since starting in Finland, it has been used in over 60 countries around the world. It was designed as a replacement for the "talk" program but has become much, much more than that. IRC is a multi-user chat system, where people meet on "channels" to talk in groups, or privately. There is no restriction to the number of people that can participate in a given discussion, or the number of channels that can be formed on IRC. All servers are interconnected and pass messages from user to user over the IRC network. One server can be connected to several other servers and up to hundreds of clients. Several larger and smaller IRC networks exist.

**Language** - The most widely understood and spoken language on IRC is English. However, as IRC is used in many different countries, English is by no means the only language. If you want to speak some language other than English, (for example with your friends), go to a separate channel and set the topic to indicate that. Similarly, you should check the topic when you join a channel to see if there are any restrictions about language. On a non-restricted channel, please speak a language everybody can understand. If you want to do otherwise, change channels and set the topic accordingly.

**Greeting** - It is not necessary to greet everybody on a channel personally. Usually one "Hello!" or equivalent is enough. Also, don't expect everybody to greet you back. On a channel with 20 people that would mean one screenful of hellos. It makes sense not to greet everyone, in order not to be rude to the rest of the channel. If you must say hello to somebody you know, do it with a private message. The same applies to good-byes. Also note that using your client's facilities to automatically say hello or good-bye to people is extremely poor etiquette. Nobody wants to receive autogreets. They are not only obviously automatic, but while you may think you are being polite, you are actually conveying yourself as insincere. If some body wants to be autogreeted when they join a channel, they will autogreet themselves.

**Behaviour** - Remember, people on IRC form their opinions about you only by your actions, writings and comments, so think before you type. If you use offensive words, you'll be frowned upon. Do not "dump" (send large amounts of unwanted information) to a channel or user. This is likely to get you kicked off the channel or killed off from IRC. Dumping causes network "burps", causing connections to go down because servers cannot handle the large amount of traffic. Other prohibited actions include:

- \* Harassing another user. Harassment is defined as behavior towards another user with the purpose of annoying them.
- \* Annoying a channel with constant beeping. (Therefore most clients cannot beep at all)
- \* Any behavior reducing the functionality of IRC as a CHAT medium.

**How to join IRC** - The first time you run mIRC you have to fill in some information about yourself (your real name, email address, nickname, IP address and Local Host name) under File/Setup/IRC\_Servers and Local\_Info, as well as the IRC server with which you want to connect. It's usually best to connect to a geographically close server. When you're new to IRC just pick a server from the prefab list. On IRC you are known to others by a nickname. You are free to choose any nickname you like, up to 9 characters long. Do not use spaces and avoid unusual ASCII characters in your nickname. It is possible you find people that use the same nickname and you may be asked to switch nicknames to avoid confusion.

**Getting started** - To join conversations, send private messages, and to handle and control mIRC you need to learn some simple commands. All commands start with a forward slash, the "/". Anything that does not begin with "/" is assumed to be a message to someone and will be sent to your current channel, or to the person you are chatting with in a private chat (see below). A list with the most used commands on IRC is given in the mIRC help file.

**Finding your way on IRC** - To join a channel, type `/join #channelname`. Try `"/join #irchelp"` or `"/join #mirc"` to give it a try... That's it! Once you get to the channel, you will see people talking. It will probably look like this:

```
<John> Hello Pat, are you new to IRC too ?
<Jake> I dont agree there :-(
<East-r> Can sb give me that too ?? I couldnt find it before
* East-r smiles
<Pat> Nope, I just have a simple question... I think...
```

Note that you will often come in during the *\*middle\** of a conversation. Unless you're familiar with the channel you may want to sit and watch it for a minute or two to see what the conversation is about. Often the channel name (for instance, `#Twilight_Zone`) has nothing to do with what conversation goes on on the channel (`#Twilight_Zone` does *\*not\** have discussion about the TV show "Twilight Zone"). So if you join `#baseball`, don't be surprised if you hear about the SuperBowl picks or even the Rock-n-Roll Hall of Fame Museum! To start talking, just type! And when you're done saying what you have to say, just hit the [return] key. You can start with something simple like "hello!". You don't have to type `<nickname>` hello! because IRC will insert `<nickname>` before all of your channel messages. In the channel's title bar you will see the channel's name and perhaps its topic. If you choose to leave a channel, just type `/part #channelname`

In the channel window that opens once you join a channel you'll see an alphabetical list of people that are on the channel on the right side of the window. Some of them have a @ in front of their name to point out they are the channel operators. A Channel Operator is someone who has control over a specific channel. A Channel Operator can also decide if control is shared or not. The first person to join the channel automatically receives Channel Operator status. Channel operators are the 'rulers' of a particular channel. This means they can kick you out of their channel for any reason. If you don't like this, you complain to them or start your own channel and become a channel operator there yourself.

**Read the help** - As soon as you joined your first channels and spent some time on IRC you will see there are a lot more commands and possibilities for you to discover on IRC. You might want to read the full version of this IRC Intro on the mIRC www pages. Almost all specific capabilities of mIRC are explained in the help file that came with the package. You just have to do the reading. :-)

[Table of contents](#)

## 5 Some short Notes and Tips on mIRC.

1. Besides that all font changes can be done from the System-menu/Font item found in every channel and query window, also a `/font` command is available to access the font settings for every window. (Did you ever use the System-menu ? Its the menu under that little horizontal bar in every top left corner of ANY window in mIRC and windows. You can also use it to save or clear the text in a text buffer as well as to save the window size and position on your screen!)
2. The Channels List dialog has improved a lot. Full Channel Name and Topic search is supported in an easily understood dialog and you can now make mIRC filter away all unwanted channels by simply setting search and suppress keys. Parents can easily filter away offensive channel list items and password protect their filtering ! If you need better kid-safety check out Kidlink at <http://www.kidlink.org/IRC/> and Netnanny at <http://www.netnanny.com/netnanny/home.html>

3. This file and several other files introducing IRC are available from the mIRC www pages in text, write, html and windows help file format. Especially files in the help file format (.hlp) can be accessed easily from within mIRC by placing them in the mirc directory. mIRC's dynamic help menu will detect them and make them available by simple mouse clicks.
4. If you lost all your right mouse click popup menu's after a version upgrade of mIRC you should check out the Tools/Popups/'Popups file:' settings !! Didn't you forget to update this line ?? Does it actually point to a valid file with popup menu settings ?? Does it point to your old mirc.ini file in c: or to the new popups.ini file in the mIRC directory ?? The same goes for aliases and remote settings that might not be set to use your aliases.ini and remote.ini files.
5. Some people experience DCC File Send problems with mIRC on a windows system with Norton Desktop installed. mIRC then suddenly shuts down completely with an error message as soon as you try to select a file to send. The problem is that Norton Desktop's feature called 'File Assist' conflicts with mIRC's DCC Send dialog. If you shut off File Assist entirely it will allow DCC transfers fine. Sometimes just disabling the 3D look in the File Assist options menu helps already.
6. The /uwho <nickname> command provides you with an improved interface with information on the person. You can store and fetch user info and addresses and view all kinds of ctcp information in it.
7. Some people get ping time-outs every time they try to connect to an IRC server using mIRC with SLIRP. This is caused by a small bug in SLIRP's version 0.95h. There is a simple patch for SLIRP that fixes this problem. You can get more info on the SLIRP www pages <http://blitzen.canberra.edu.au/slirp> (I heard version 0.95j exists ?)
8. Some people experience unexpected behaviour of the ON CHAT and ON SERV event. They try to use it for higher level users which is in essence NOT possible. You cant use these events above level 1 (or whatever other default you set). This is caused by the way of handling DCC's totally separated from the other messages. DCC messages are not handled by servers... remember ??

[Table of contents](#)

## **6 FEATURES, TIPS and ANSWERS to QUESTIONS.**

### **6-1 How do I run the 32-bit mIRC on Win 3.1 or WFW 3.11 ?**

To be able to run 32-bit programs on Win 3.1 or WFW 3.11, you need to install the win32s 32-bit extentions. Not all 32-bit programs can function on this small 32-bit base, but mIRC can !! Make sure you have installed a recent version of these extentions.

Contrary to what I thought earlier, you don't need to have a 32-bit WINSOCK installed. mIRC will work fine on the Microsoft TCP/IP-32 winsock which is 32-bit (but doesn't have modem support as far as I know); also, using the popular Trumpet winsock (-at this moment- only available in a 16-bit version) will work. (DONT misunderstand... I'm talking about win 3.xx only here!)

On WFW 3.11, problems can occur with the win32s extensions due to versions older or newer than the win32s version 1.30 used with the compiling of mIRC32. Problems with the win32s extensions can cause mIRC32 to crash on startup or give all kinds of strange behaviour on startup. It's best to use the 1.30 version of win32s. Also, if you experienced the problem that your toolbar buttons in mirc32 turned black you might have to download the new win32s version 1.30 revision 166 or later and install it.



Microsoft: <http://www.microsoft.com/kb/softlib/mslfiles/pw1118.exe>  
NCSA: <ftp://ftp.ncsa.uiuc.edu/web/mosiac/windows/win31x/win32s/ole32s13.exe>

### Table of contents

## **6-2 How do I run the 32-bit mIRC on Windows95 ?**

On Windows95 no problems running mirc32.exe have to be expected. If you can't manage to get it running you might have a 16-bit winsock. Contrary to the case with Win3.xx, you CAN'T run mirc32.exe on Windows95 with a 16-bit winsock ! The most common error message then is "Error: Cant resolve local host name". So, if you have problems running the mirc32.exe, while the 16-bit mirc.exe works fine, MAKE SURE that you use a 32-bit winsock. The in Windows95 built in winsock is 32-bit. The famous Trumpet winsock is NOT 32-bit !! (Trumpet recently released a beta of a 32 bit winsock. mIRC is not tested on this program)

If you are using Windows 95 and a 16-bit winsock like Trumpet, Spry, Chameleon, etc., there is a known communication problem between 32-bit programs running in the 32 bit protected system and the 16-bit subsystem with the winsock in it. Your 16 bit winsock, like all 16-bit applications, runs in a 16-bit subsystem under Windows95. Under Windows 95, 16-bit winsock.dlls don't have access to the 32-bit protected system and vice versa, so 32-bit applications can't use the winsock's TCP/IP services. You must install a 32-bit winsock.dll to solve this. Preferably the native Windows 95 TCP/IP(winsock) support.

To use the 32-bit version of mIRC, you must install the 32-bit stack and dialer that are a part of the Windows 95 installation. The dialer then needs to be configured to access your Internet Service Provider. At windows95.com (<http://www.windows95.com/tcp.html>) you can get detailed information on configuring your Windows 95 TCP/IP software to access the Internet. Netcom has provided their own dialer FAQ (<http://www.netcom.com/faq/dial95.html>)

### Table of contents

## **6-3 How do I solve the "Can't Resolve Host Name" error?**

If you are using Windows95 first find out if you're trying to run mIRC32.exe with a 16-bit winsock. You need a 32-bit winsock to run the mirc32.exe. If you have a 16-bit winsock use the 16-bit mirc.exe. Read section 6-2.

In the File/Setup/Local\_Info dialog box, mIRC attempts to guess your IP address. But if you don't have an actual IP address (such as with TIA, Twinsock, SLiRP, or some other SLIP emulator), or if your stack is strange or there is some network oddity, it might not fill these in correctly. To correct this, uncheck "On connect, always get IP Address and Local Host" and manually enter your IP number. Also fill in the 'Local Host' by hand. If using TIA, Twinsock, SLiRP, etc., simply enter your provider's local host name and IP, not your own. (Your provider's IP is not 192.0.2.1 or anything like that! That's a dummy IP used by these emulators. If you don't know your provider's local host name and IP, ask!)

Another possibility is to make mIRC use the Quick IP method instead of the normal one, but this will fail if your local settings in win.ini or registry file are not ok. It's also possible that mIRC cannot complete the reverse look up it tries and gives the "Can't resolve host name" error when your IP name is not properly configured in your providers DNS. In that case you should ask your provider to assign a IP name to your IP Number to solve the problem. For your provider setting up the IP name on the DNS is not that hard, and doesn't take much time.

Note: You must be disconnected from any IRC server before changing these settings.

Also setting the Ident server to active might help !! (See section 6-7)

[Table of contents](#)

## **6-4 How do I solve the "Unable to resolve IRC server name" error ?**

If you can't get mIRC connected to one particular IRC server and you get this error you should first give another server a try... If you can connect to any other server you should check if you didn't make a typo in the setup of your initial server address.

(File/Setup/IRC\_servers/Edit)

If you get this error with whatever IRC server address you try, it is very likely your provider's DNS (Domain Name Server) is down, malfunctioning or very slow. Especially if mIRC always worked flawlessly for you. Besides waiting or giving your provider's helpdesk a phone call you can't get this fixed.

mIRC needs a DNS to translate (resolve) the IRC server's IP Address you specified, into an IP Number. (Like the IP irc.law.emory.edu resolves to the IP Number 170.140.50.195 ) (To see this give the command "/dns irc.law.emory.com" in mIRC) You could bypass the need for the DNS lookup by specifying the IP Number in your server setup instead of the IP Address for every server.

Under File/Setup/IRC\_Servers/Add or Edit server, specify your IRC servers like :

Description : irc.law.emory.edu (Emory University)

IRC Server : 170.140.50.195

Port : 6667

If nothing helps you of course could also have a rotten or miss-configured winsock?

[Table of contents](#)

## **6-5 How do I solve the "Not enough user parameters" error?**

You will get a "Not enough user parameters" from mIRC if you try to connect to a server but you didn't have the local host filled in, or had it filled in incorrectly. Check your entry under 'File/Setup/Local\_Info/Local Host.' Also check to see if you supplied mIRC with a valid E-Mail address under 'File/Setup/IRC\_Servers/E-Mail:'. (See section 8, too.)

Note: The e-mail address is used internally by mIRC and cannot be seen by others when they do a /whois on you. Some have noticed the "email:" field in the User Central dialog (/uwho command). This field is made up of your or the other person's user@host (your username plus your local host name)... in many cases you'll notice that it doesn't reflect the actual e-mail address. It has nothing to do with 'File/Setup/IRC\_Servers/E-Mail:'. Enter the correct address there!!

[Table of contents](#)

## **6-6 How do I solve the "You haven't registered" error?**

Note: mIRC is freeware and you can't even register it .. This is a server message to you..

If you receive this message and/or you get disconnected that quickly, your Local Host name and/or IP number might be wrong, or not filled in at all. Look under 'File/Setup/Local\_Info/', and check if the local host is correct and if the IP address (number) is filled in automatically... An easy way to solve a wrong IP Address is setting the 'On connect, always get' "IP Address" and "Local Host" to "ON" and restarting mIRC. If this doesn't help you could be trying to connect to a NON-public server!! Try another server to check this...

## Table of contents

### **6-7 Why am I unidentified and what does it matter ?**

An ident server normally is maintained by the Unix machine of your network provider... It is a kind of nameserver that guarantees your Identity. Since most standalone windows machines are not correctly backed up by an ident service, such a server is built into mIRC. IRC servers can do an Ident request to your Ident server and then expect a standardized kind of answer. More and more IRC servers require you to be identified in some way, and they will disconnect you if you're not identified ! Also if you don't react, or do so in the wrong way, they can decide to disconnect you.... You can check if you're properly identified by doing a /whois on yourself. The first line in the reply should NOT contain a ~ (tilde). If you have a ~ in it try activating the Ident server.

mIRC's built in Ident server can be switched to active under File/Setup/Identd. Set it to :

User ID: <account name>	(The part before the @ in your E-mail address normally)
System: UNIX	(ALWAYS fill in UNIX !! not dos, win or *whatever* else !!)
Listen on port: 113	(The standard ident port number)

A problem that TIA and similar SLIP emulator users will experience is that, despite checking the Ident Server to active, mIRC will never reply to an Ident query. This is because with TIA it never gets the ident request. The slip emulator won't pass through the Ident request from the IRC server to mIRC... That just means you'll be seen as nick!

~account@machine.net which is not the end of the world, unless you happen to be using a server that requires an Ident reply and disconnects you... This can't be solved by mIRC or any other client..... You will need to find another server OR get your provider to set up proper indenting...

## Table of contents

### **6-8 I can't get mIRC to DCC send or initiate a DCC chat! Getting files works fine..**

Read the paragraph that may apply to you..

In almost all cases that DCC sending and chatting mess up, the initiating party causes the troubles. In almost all cases DCC Send or Chat stops functioning (suddenly) this is caused by wrong Local Host and/or IP Address settings at the senders side. (The Local Host and IP Address settings are found in the File/Setup/Local\_Info menu.) Typically in the case that your IP Address is not correctly set, the other party will get your dcc message and then try to confirm and connect to your incorrect IP Address. Of course, this doesn't work and you both sit there waiting.

A good cure to solve these DCC problems is to start with this :

- 1) Disconnect from your IRC server.
- 2) Clear the current text in the Local Host and IP Address boxes under File/Setup in the Local\_Info dialog.
- 3) Select both 'On connect, Always get' Local Host and IP Address options in the Local\_Info dialog.
- 4) Set IP method to Normal.
- 5) Leave the Local\_Info dialog and the File/Setup menu both with 'OK'.
- 6) Reconnect to your IRC server.

If this doesnt work experiment with the 'On connect' .. settings... !! For instance, try to set your Local Host address by hand and let mIRC look for only the IP Address on each startup. If

you still cant get it to work try to use the Quick IP method instead of the default Normal one. The Quick one looks up the address locally from your machine instead of from your providers Domain Name Server. Read more about these cases in the mIRC help file !!

Everybody...

Check your time-out settings! Make sure that your time-out values in DCC/Options are set large enough!! "Get/Chat Dialog time out after" and "Send/Get Transfer time out after" are recommended to be set to at least 60 and 120, respectively.

In case you never ever managed to get sending files or initiating a DCC Chat to work it might be your provider blocks these DCC connections by the kind of internet access they give you.

If you use TIA or Twinsock or Slipknot ...

If you use TIA (The Internet Adapter) or Twinsock, at this point you cannot use DCC send or initiate DCC chat (with any IRC client, not just mIRC). You may want to try SLiRP or vTCP. SLiRP was the first SLIP emulator to allow DCC sending and initiating DCC chat. (As you know DCC get should always work fine, whatever connection you have. Besides firewall blocking you that is.) Virtual TCP is tested and proofed to allow DCC sending and chatting. More info <http://blitzen.canberra.edu.au/slipr> and <http://www.infoexpress.com/vtcp.html>

If you use SLiRP ...

With SLiRP (currently running 0.95j) set File/Setup/Local\_Info/ 'always get local host' on connect to Active, IP Address should be the fake IP used for SLiRP (10.0.2.15 usually). Then DCC Send, Chat, and everything else should work perfectly fine, even on Windows95 with the Dial-Up Networking. (thanks Dave!)

If you have Dynamic IP..

If you have dynamic IP (your IP address is different each time you log on), make sure that "On connect, always get:" in the File/Setup/Local\_Info dialog is set to get the Local Host and IP Address. If these were already set to ON make sure the correct 'local host' name and 'IP Address' are found by mIRC... on some winsocks this is rather tricky... If you have a non-compliant stack, mIRC may not be able to correctly find your local host (domain name) and IP. With dynamic IP addressing you are in trouble then !! DCC file sending and initiating a DCC Chat (contrary to file getting and accepting a DCC Chat) requires that mIRC knows your correct IP number. Even without an IP number at all, mIRC will work as far as normal chatting is concerned, but won't allow DCC file sending or initiating a DCC Chat.

If you have Static IP..

If you have a non-compliant stack, mIRC may not be able to correctly find your local host (domain name) and IP. In the File/Setup/Local\_Info dialog, uncheck the options to "Always get the 'Local Host' and 'IP Address'" and manually enter your correct Local Host and IP.

A known Windows95 bug causes a lot of people to report that mIRC (and any other IRC program) gets/finds the old (now wrong) Local Host name and/or IP Address after switching Internet provider. This blocks their capability of DCC Sending files and Initiating DCC Chats. If, for some reason, no matter what you do, mIRC picks the user ID (Local Host name) from the Internet Service Provider that you no longer wish to use this is fixable by editing the registry. If you open Regedit and look at MyComputer /HKEY\_LOCAL\_MACHINE/System/CurrentControlSet/Services/VxD/MSTCP you will see the Domain and NameServer fields from your old provider. These fields will persist even if you uninstall Dial-Up Networking and re-install and go through the TCP/IP settings again ! The best way to solve the described problem is going to Start/Settings/Control\_Panel/Network/

double click on TCPIP/ select DNS\_Configuration/ and set the HOST field to the hostname (ID) you have on your new provider.

If you use Windows95 mIRC allows you to send long file names with spaces in them, but other IRC programs very often cant handle this. This might cause your transfers to fail. You might want to select mIRC's option to fill (up) spaces in such a long file name by an underscore.

[Table of contents](#)

## **6-9 I can't get mIRC to DCC Get!**

If you cant get files by DCC from somebody -he/she- might very well (almost sure) have the problem described above in section 6-8. Especially if you can get files from others fine. Ask the sending party to check his IP Address settings.

It sounds stupid but also full harddisks block getting files very effectively !

Some people also get the error "unable to create socket" on Internet in a BOX. If this happens, try to connect again (choose File/Connect). This sometimes works.

[Table of contents](#)

## **6-10 How do I use Copy and Paste?**

In channel, query and DCC chat windows mIRC uses a nice select-and-copy-in-one-move feature. Copying text in those windows is done automatically as soon as you release the mouse button you used to select the text with. Just highlight the text and release. Isn't that a nice feature? Use ctrl-v to paste the copied text. The normal Windows' "Copy" menu is not just left out due to ignorance, but is due to the use of colored text in the graphical windows in mIRC. Windows does not support colored text in standard text boxes, and Khaled had to write all text box/windows routines himself! :-) And since using cut or paste in a channel, query or chat window makes no sense anyway, he was able to invent and implement this feature. All text shown in mIRC outside the channel windows (in settings, menu's, dialogs and on the edit box line) can be copied, cut and pasted normally using the ctrl-c, ctrl-x and ctrl-v key combinations.

[Table of contents](#)

## **6-11 How do I get colored text in mIRC?**

mIRC has a nice feature that displays normal text, highlighted text, actions, and server messages in different colors (black, brown, purple, and green respectively). To see the colored text, make sure that File/Options/Extras/'Windows colors' is -NOT- selected. In this case, all of Windows' colors (as set in the Control Panel) are used in mIRC except for the channel background color. (You can imagine what otherwise would have happened with purple lines on a purple background, eh?) If you set this option to "Windows colors" instead, mIRC will use all of Windows' colors, but you will lose the multi-colored text for actions, etc. Also make sure to set File/Options/IRC\_Switches/'Actions are purple' to active.

[Table of contents](#)

## **6-12 How can I customize the fonts used in mIRC ?**

From mIRC 3.9 on the Misc/Fonts menu has been dis-continued. All font settings can be done

from the windows' System Menu's (those menu's under that little horizontal bar in every top left corner of ANY window in mIRC and windows) or the /font command. Use /font or select the 'Font' option in the System menu and a fonts selection dialog will pop up. Here you can choose whatever fonts are available on your machine. This could be New Times Roman, Arial, MS Sans Serif,..... whatever you like. You can also set the font size and bold. (italic is available but disabled for most fonts at the moment) The settings you set here will be saved as the default for the window you are in.

[Table of contents](#)

## **6-13 How do I get mIRC to use 3-D dialog boxes?**

Recent mIRC versions automatically use 3D dialog interfaces. All you have to do is provide the proper ctl3d DLL(s). The dialog boxes will take on a nice gray color with a nice slightly "chiseled" appearance. The normal channel windows and or text wont be affected by this 3D look.... To see the 3-D dialog boxes in mIRC16, you first have to make sure that you have the file CTL3DV2.DLL correctly installed. CTL3DV2.DLL was not originally included in Windows or Windows for Workgroups 3.11, so you may not have a copy. Check your windowsdirectory. For proper installation, you should have only one copy of this file, the newest one, on your system, in the system directory. (Having multiple copies of the ctl3dv2.dll will mess up the behaviour of this dll !! Check your ENTIRE harddisk !!) You can FTP the file from ftp://bitsy.mit.edu/pub/dos/alpha-beta/techinfo/ctl3dv2.dll  
In mIRC32 you need the CTL3D32.DLL. It has to be in the c:directory. Again make sure to have only ONE copy of this DLL on your entire Harddisk .... On windows95 make sure to have version 2.26 or later of this DLL !! Also WFW311 people can use this one. (You can see the version by using the EXPLORER, select the file, right click, select properties, version.)

[Table of contents](#)

## **6-14 How do I specify people correctly in the auto-op and ignore list ?**

In the Auto-op and Ignore sections under File/Options/Control/.

If you don't care about channel takeovers and other annoying things, you can just specify the nick of the person you want to auto-op or ignore. If you want to be sure you 'op' or ignore the right person for ever, specify them by their address if possible.

The full address format is Nick!Account@machine.

Lets assume that Bill responds to a "/whois bill" with:

```
Bill is gates@pentium.company.com * Billy A Lot Of Names Gates
Bill @#mIRC
Bill irc.server.company.com (A superb company)
Bill 56 seconds idle
Bill End of /WHOIS list.
```

In that case, Bill would be best referred to by

```
bill!gates@pentium.company.com
```

You could also use \*!gates@\*.company.com if he works on several machines with different nick's.... To add a user to the lists use the /auto and /ignore commands (like /auto \*!gates@\*.company.com), or add a user by simply typing him into the specific edit box. Make sure to switch the functions to active by the checkbox under File/Options/Control totally right

of the function markers of the edit box. Or use the commands `/auto [on|off]` and `/ignore [on|off]` on the command line.

You can make the auto-op and ignore more specific by specifying some parameters.

The auto-op can be set to make people operator only on certain channels if you want. In that case just specify the channel names. `/auto <nickname|address> [#channel1,#channel2,...]` If you dont specify a channel name mIRC will op the person on all channels where you're operator.

The ignore can be set to ignore only certain actions from a person. mIRC distinguishes between private messages, text in channels, notices send to you, ctcp commands and invitations. From the command line you can use: `/ignore [-pcnti] <nickname/address> [type]` where p = private, c = channel, n = notice, t = ctcp, i = invite. (more parameters discussed in the help file!) In the ignore section an ignore will look like `nick!userid@host.domain,private,channel,notice,ctcp,invite`. If you dont specify any of these additional parameters mIRC will just ignore somebody totally.

[Table of contents](#)

## **6-15 How can I make mIRC react to Remote Commands?**

Use the Tools/Remote/Commands section.....

mIRC can react to CTCP commands given by remote users. You can customize your responses to CTCP commands and offer files: features that usually require scripting but here are handled by the Tools/Remote/Commands section. It is this section in mIRC that can handle OPME, KICK, MYLEVEL, or XDCC SEND commands for you, if you configure it well. Setting up the remote section is not an easy task, but the way it works guarantees full freedom to make it do what you want it to do.

As soon as your mIRC receives a CTCP command from somebody else, it checks if that command is defined and if so, it matches the required user level against the level of the remote user. If the remote user has a suitable user level his command is executed... All commands given to you have to have the format `'/CTCP <yournick> <command> <parameter(s)>'`. If a command is defined on several levels, the highest ranked one is executed. See the help file included in the mIRC39.zip package for detailed instructions and read the tutorial in section 7 of this FAQ.

[Table of contents](#)

## **6-16: How can I make mIRC react to Events?**

Use the Tools/Remote/Events section..... Events are all things happening on channels or in private conversations on IRC. People joining, leaving, getting opped, deopped, greeting you, even plain talking all are Events... You can configure mIRC to react however you like to almost anything that can happen in IRC. See the mIRC help file for further instructions and read the tutorial in section 7 at the end of this FAQ.

[Table of contents](#)

## **6-17 How do I specify my friend Bill correctly in the Users list with an access level of 2?**

(In the Tools/Remote/Users section..)

You can add your friend Bill with access level 2 to the Users list in two ways:

1. Specifying the nick -- "2:bill"  
and/or by...
2. Specifying the address -- "2:bill@gates@pentium.company.com"

Of course, specifying the address is the safer option. Wildcards are supported, such as "2:\*!gates@pentium.company.com". In which case all nicks bill could use on his account are recognized. To get the ON OP, ON DEOP, ON SERVEROP and ON NOTIFY events working you \*have\* to specify a person by his nick !! It won't work if you specify his full address only... In the case of bill, you could use BOTH Users list lines mentioned above. You could just type the nick and/or address straight into the Users section under Tools/Remote but mIRC also has special commands to add people to your Users list straight from the command line. Look in the mIRC help for the commands /auser, /guser and /ruser.

[Table of contents](#)

## **6-18 I get disconnected after the /LIST command.**

It is quite common that everytime you do a /list, you get disconnected part way through. Even if you try a partial list, eg. "/list #love", it runs for about 90 seconds and then disconnects you. Sometimes using a local IRCserver helps.

You didn't do anything wrong... It is important that you realize what happens if you do a /list command. The server generates a list of all channels (about 2000 on Efnet) and quickly sends that list to you. BUT, because the data throughput isn't infinite, the whole output of the /LIST command is queued in a buffer on the server. At some point that buffer gets overrun, and the server, detecting this ('reached maxsendq'), disconnects you. This mechanism is made to disconnect people who generate more characters per second than any 'normal' person uses for normal conversations. It's a protection mechanism, but unfortunately the server does not recognize that the data stream you caused is simply the result of your harmless /list command, and disconnects you.

You can also tell mIRC to show only channels with a minimum and a maximum number of people. Also, if you specify a #string, then mIRC will only list channels with that string in their title. BUT, for the problem described, this does not matter/help at all!! The server ALWAYS returns the entire #channel list and mIRC (like all other clients) takes care of the filtering!! So, asking for a partial list to prevent you from disconnecting won't make ANY difference!!

So, this isn't an mIRC bug. It's a pity, but there's nothing you can do to prevent this from happening. It's just one of all-too-many IRC oddities. Try using some other servers, as many aren't quite so touchy. Oh ... check out the list window popup menu !! Once you've done a full "/list" the entire list is stored and available for re-processing with keys and min/max settings! Right click in the channels list window and play with it!

[Table of contents](#)

## **6-19 Running multiple copies of mIRC.**

mircc.exe 16-bit :

You can't just start the 16-bit mircc.exe a second time. If you want to run a second copy of mIRC16 (don't ask me why), it's best to rename your mircc.exe to mircc1.exe and also copy your mircc.exe to another instance, like mircc2.exe. ( In the windows File Manager select mircc.exe and with File/Copy copy it to mircc2.exe, and with File/Rename rename it to



mircl.exe. Both copies can be kept in the same directory (the mIRC home dir.) Do the same for the mircl.ini file that can be found in the c:\windows directory or in mIRC's home dir. (Unless specified otherwise mIRC uses the mircl.ini file found in c:\windows. ) Copy mircl.ini to a mircl1.ini and to a mircl2.ini. Now in the Program Manager, rename the mIRC icon to an icon called mIRC1, and use that to start the first copy. 'Under' the icon, on the startup line, state "c:\path\mircl1.exe -i c:\path\mircl1.ini" to make sure mircl1.exe will use the mircl1.ini! Also, make an icon called mIRC2, and use that to start the second copy. On the startup line 'under the icon' state "c:\path\mircl2.exe -i c:\path\mircl2.ini" to make sure mircl2.exe will use the mircl2.ini! The -i startup line parameter tells mIRC to use the file after it as ini file. If you want to place the mircl.ini files in another directory than your windows or home dir, make sure to specify the full and correct path! You can now startup both exe's and configure them independently..... Nice huh?

mircl32.exe 32-bit :

If you use the 32-bit version of mIRC you CAN run a second instance without problems without having to provide a second, renamed .exe file. You don't have to rename and or copy the mircl32.exe. BUT you still have to create 2 icons with unique mircl.ini names on the startup-line to get your 2 mIRC's working independently and not interfering .... Don't forget that :-)

[Table of contents](#)

## **6-20 How do I make myself invisible to other users ?**

And what it does and doesn't do. ;o)

In mIRC, you can make yourself invisible with: `"/mode <your_nickname> +i"`. The title of the STATUS window will show (+i) after your current nickname. If you change nick, you will remain invisible under your new nick. Use `"/mode <your_nickname> -i"` to make yourself visible again.

Nobody can detect your changing from invisible to visible or vice versa. When you're invisible, you are always visible to all the people who are on the same channel(s) as you are. When you join a new channel (while invisible), everyone in the channel sees you joining. Also, leaves are NOT hidden. Also, when people do a `"/whois <yournick>"` they will see a normal whois list on you if the nick they enter matches your's exactly.

So, what's the point of this "invisibility" capability ?

Users NOT on the same channel as you WILL NOT be able to see your name if they do `/who <#channelname>`. Also, if they do `/names` to list all the people currently on IRC, they won't see you. Also doing `"/who *part.of.your.address.net*" won't result in your nick turning up. The point of being invisible is not to be able to stroll along channels like a ghost without anyone seeing you, but to hide (a bit) from users that scan channels with /who to find you. If they scan channels by actually joining them, then invisibility won't help you. To get rid of annoying people: first make yourself invisible ... then change your nick and make sure those people can't join the channels you are on in order to find out your new nick. (also, doing: /ignore <other_persons_nick> helps a lot !`

[Table of contents](#)

## **6-21 How does the sound command in mIRC work ?**

Assuming you have a sound card or a speaker driver allowing you to play wav files on your PC, you can make some funny use of sound on IRC with mIRC. Already available in mIRC

before version 3.7 was the /wavplay <c:\path\sound.wav> command which allowed you to play a .wav sound file locally to you... Now the /sound command allows you to send a request to any other party to play a .wav file he and you both have...

First set mIRC to Accept sound requests under File/Options/Sound Requests/ The command syntax is /sound [nickname|#channel] <filename.wav> <action text>. As you see in the play request an action text has to be specified which will display on the other side... I have to stress that the actual wav file is NOT sent to the other party. We dont want to stuff the net with data you know :-) The command just triggers wavplaying at your and somebody elses machine..... So make sure that the other party has the wav you want to start...

A "/sound friend tada.wav does a tada" command done by you will result in '-> \*friend\* does a tada' and the tada sound at your side (the \*friend\* confirms the destination to you) and the action "\* thisguy does a tada" with the tada sound played at your friends side... This command currently ONLY works in between mIRC users !! The command format is /sound [nick/channel] <file.wav> <message>. The message will show as an action to the recievers and both you and they will hear the wav you selected. mIRC will look for your wavs in the directory you set under Misc/Options/Sound Requests/ Playing with sound you might like the /speak command too ! This command makes mIRC use the Text To Speech program Monologue (by DDE) to speak up whatever text you add to the command .... Monologue was sold with older soundblaster cards. Read section 6-25.

[Table of contents](#)

## **6-22 How does the WWW support in mIRC work ?**

In mIRC 3.92 Netscape, Mosaic and MS Internet Explorer are supported. You can now surf the www with your friends and exchange addresses of the best pages you find. mIRC's URL catcher automatically stores URL's from text passing by in channels, privates and topics and stores them in a listbox. You can also send one or more URL's to the channels you are on (or to any private conversation you are in) by the URL window popup menu. (Popup menus are activated by the right mouse button !!) You can store URL's for future reference and set them in any order you want by the use of markers. Newly found items are appended to the end of your URL list with the '?' marker. When the marker for a URL is changed to something else it is sorted into the permanent list. In the URL options dialog box you can set mIRC to delete '?' marked items on exit. Also a new string is created which represents the page your Netscape currently shows. You can use this \$url in popup menu definitions and in remote definitions.

To get the hang of it first enable the URL catcher in the File/Options/URL-Catcher/ menu. Also fill in the place where your www browser can be found. From that moment on mIRC will find and store all www addresses passing by on the channels you are in. You can see this going on if you open the URL window (use the URL button on the Toolbar or the /url command). Then if you want to view an URL in your www browser just highlight it in the URL window and select 'view' from the URL window popup menu (right mouse button). Alternatively you can set mIRC to react to a double click on an URL in the URL window. Once you start to use the URL catcher you'll get the grip of it... Have fun !

[Table of contents](#)

## **6-23 What is the built in finger server for ??**

Lets first explain that there are 2 sorts of fingering on IRC. First there is the IRC finger that almost all clients can handle. This is a CTCP command with the syntax '/ctcp nickname finger'. The reply you will get to this CTCP command is usually a one liner set by the user

you do the finger on. In mIRC you can set this reply under File/Options/Action\_Lists/ at the Ctcp\_finger\_reply line. Some other clients use the alias '/finger nickname' to shortcut this CTCP finger command but thats not the official way..

The other finger command comes from the finger system familiar to UNIX users. On UNIX systems you can finger the address of another user to find out some more about that person. Normally this fingering will deliver you the persons home address, telephone number or other usefull info. This info is given to you by a central finger server and you need a finger client to get it. mIRC already has a finger -client- built in for a long time under Tools/Finger. If you specify a users address, mIRC will try to get some info on the person at the other end. The '/finger <nickname|users\_address>' command is doing the same thing. (If you specify a nickname mIRC will try to find out that users address and finger it.) Mind the syntax difference from the IRC finger command !!

In the newest mIRC, a finger server is built in to enable you to provide finger info even if your school, provider or company doesnt maintain a central finger server. Under Misc/Options/Servers/ you can enable this Finger server. You also have to provide a text file with the info you want to give. This file should meet a certain syntax as explained in mIRC's help file. A last remark; dont be surprised if you find that the UNIX finger is hardly used on IRC !

[Table of contents](#)

## **6-24 How can I easily serve my collection of shareware to others, using mIRC ?**

As a lot of people on IRC promote and share the best they found around on the net, mIRC now offers a unique built-in Fileserver. This Fileserver feature is somewhat of a cross between DCC and FTP. You open the server window to someone, (it's a special DCC chat window), restricting them to a certain directory tree, and they can browse your file listings, change directories, read text files, or get files.

The syntax to set up a DCC server connection to somebody is:

```
/fserve <nick> <max gets simultaneously allowed> <homedir> [welcome file]
```

"Max gets" is so that the other person doesn't bring down your machine with too many parallel gets. 4 is probably a reasonable number. The other person will have access to his homedir and all dirs DOWN in the directory tree from that homedir. "Welcome file" is a text file you can write and specify that will welcome users to your file server. It's optional. Examples:

```
/fserve Krejt 3 c:\temp\serve c:\temp\serving\welcome.txt  
/fserve Mookies 2 c:\outgoing c:\network\mirc\welcome.txt  
/fserve Friend 7 c:\
```

Keep in mind that you can't set up a server to yourself... you need others to test your server..

Typing help in the file server will show the available commands, which are styled after Unix and DOS. "ls" or "dir" will show a directory listing, for example. Even switched commands like "ls -k" (show file sizes in kilobytes) and "dir /w" (show a wide directory listing) work. The server supports all normal ftp commands like cd <dir> , cd., dir, ls, get, .... but NOT put, hash, upload etc. There is no possibility to delete files in a server connection. Safety risks are none or minimal due to the major restricting of available commands.

Of course, the /fserve command can be used in your Remote section....

Set up a simple Tools/Remote/command like :

```
1:server:/fserve *1:$nick 3 c:\temp\serve
```

Set the commands to active (/remote on) and off you go....

Other people only have to type "/ctcp yournick server" to activate the server. You can't set up a server to your own mIRC!! So, others have to test your server !! In the directory c:\temp\serve, you place all files other people are allowed to get from you. The people using your server will have access to the c:\temp\serve directory AND ALL directories BELOW it.. like c:\temp\serve\games.

[Table of contents](#)

## 6-25 How do you use Monologue's Text To Speech support ??

mIRC uses the software package 'Monologue' to speak out text at your wish. You can make it to say out loud whatever you want that happens on IRC; on channels or in private chats etc. Monologue was sold in a software package with older Soundblaster sound cards. Monologue is not for free and not shareware either ! It is made by the Company First Byte (<http://www.firstbyte.davd.com/html/fbinfo.htm>). The software is some years old but I heard a 32 bit version is in development and also the 16 bit version works rather well.. and a lot of people have it. Therefore mIRC supports it to have Text To Speech support.

First you have to make sure Monologue is properly installed on your PC and working in good order. It has a built in test function and should work properly with the test and clipboard support before trying it from mIRC. To use it with mIRC you have to enable monologue's DDE support. Through DDE mIRC communicates with it. Once you enabled the DDE support you can fire up mIRC and give it a try.

In mIRC the '/speak <text>' command is used to send text to Monologue. It is used like :

```
"/speak this is a test"  
"/speak can you hear me ?"  
"/speak this is fun"
```

It is fun indeed ! Once you manage to speak one-liners from a mIRC window by this /speak command, it is not that hard to get it speaking whole channels or conversations ... assuming you can handle programming mIRC's remote events section. (Read and study section 7.) It is not meant to be a plug and play solution ;o)

[Table of contents](#)

## 7 SOME NOTES ON 'PROGRAMMING' mIRC (Tutorial?)

I never thought this section was gonna grow into what it is now... It proves to be maybe the most important part of this FAQ. You all clearly showed me there is a need for this Aliases, Popups and Remote tutorial. I hope this section will grow on into a nice mIRC reference (and teaching) manual. Have fun with it !

mIRC has 4 sections which can be "programmed" in some way: The Aliases, Popups, Remote Commands, and Remote Events.

**The best way to learn and understand the way these sections work is by studying them in consecutive order. Really! Please don't try to take a shortcut to the Remote sections before you really understand the Aliases and Popups. The lessons you'll learn from each are cumulative.**

(A note for those familiar with ircii: mIRC's aliasing syntax is similar to ircii's, but not exactly)

the same. mIRC's Remote Command and Event sections allow you to implement many of the same things you can do with ircii scripting, but the structure and syntax are a bit different.)

[Table of contents](#)

## 7-1 The Aliases section.

Under Tools/Aliases... the Aliases section will be found.

In this section you can define shortcuts for often-used commands such as join, part, quit, opping and deopping, and whatever else you want. Even things you frequently say can be placed in an alias... In general, ALL commands which can be used in the edit box (the line or lines you normally use to talk/type in) can be used in an alias. Aliases can also be used to execute several commands at once. You can even define aliases that are launched by the function keys F1 - F12.... (Finally some use for those never touched keys..)

From mIRC version 3.7 on you can even call an alias from popup menu's and remote lines.

### A Simple Example: /join

OK, let's start right off the bat with an example. To join a channel, you normally have to type:

```
/join #<name>
```

...where #<name> is the name of the channel, of course. Since this command is used so often, it'd be nice to use it by typing as little as possible, right? So let's create an alias for "/join #<name>" -- let's create a shortcut.

Instead of typing "/join #<name>", let's make it so that we can type "/j #<name>" with the same results. In that case, the Alias would be:

```
/j /join $1
```

(You don't have to understand exactly what that means to mIRC at this point, just what it will do. I'll explain the alias syntax shortly.) Give it a try by placing the line "/j /join \$1" in the Aliases list under Tools/Aliases (if it isn't already there). Then try it out on the command line (where you normally type) with a simple command like:

```
"/j #test".
```

You'll see that "/j #test" now does exactly the same thing as "/join #test" -- it joins you to a channel named #test. This construction is called an alias.

### Alias Syntax.

All aliases use the format /<aliasname> /<command>

In other words, first your "shortcut" name (like "/j" in the example above), then a space, and then the actual command your shortcut is pointing to (like "/join" in the example above).

But it doesn't stop there. Aliases also use a number of special STRINGS. There are three pre-defined strings (#, \$me and \$!) which always mean the same thing to mIRC (new predefined strings are the \$read [-l#] <filename.txt> and \$url .... I'll discuss them later too.), and then several variable strings (such as \$1, \*1, \$\$1, \$?, # \$1, # \$ \$1, and even # \$?).

Don't worry, we'll explain all of these in good time. Right now, let's concentrate on \$1. After all, that's the string we used in our example above:

```
/j /join $1
```

\$1 means "the first word typed after the Alias, whatever that may be." So if you type "/j #test", mIRC knows that this is a shortcut for "/join #test". If you type "/j #hello", it will know that this is a shortcut for "/join #hello".

### **Multiple Commands.**

Now, let's say that you'd like to join several channels at once. You can make an alias to do this. Just separate the commands with the | character (on most keyboards, this is the shifted version of the backslash key)... Let's say that you'd like to be able to type "/j3" and then follow it with the names of three channels, and they'll all be joined at once. The alias would be:

```
/j3 /join $1 | /join $2 | /join $3
```

So if you placed this alias in Tools/Aliases and you were to type:

```
/j3 #test1 #test2 #test3
```

...mIRC would "/join #test1", then "/join #test2", and then "/join #test3".

In this case, you see that \$1 means the first word typed after the Alias. And \$2 of course means the second word, and \$3 the third. Now, mIRC is obviously expecting three words -- your channel names. So what would happen if you just typed "/j3 #test1"? mIRC would complain about the missing parameters, as it's expecting more channels to be named. There is a solution to this problem, the \$\* string, which will be explained later.)

### **Example: /say**

Let's say that there's something you find yourself typing into a channel all the time, like your home page address, a greeting, etc. Take a look at these alias examples using the "/say" command: "/say" simply types the indicated text into your current channel or private message window.

```
/hello /say Hello to all of you...  
/www /say The mIRC page is at http://metro.turnpike.net/mookies/mirc.html  
/lines /say This is the first line | /say This is the second line | /say This is the last line
```

The last examples show how to /say multiple lines of text in one alias. Just separate the commands with the | character. You also see that not all aliases always have Strings in them.

### **Example: /me**

Actions in an alias...

```
/fun /me thinks this is fun!  
/hug /me hugs $1 firmly!
```

The second example uses a string, \$1. If you were to type "/hug everybody", this would be equivalent to "/me hugs everybody firmly!"

### **Example: Shared Strings**

Multiple commands can also share the same string.

```
/jtest /join $1 | /topic $1 just testing
```

In this alias definition the parameter \$1 is used twice. If you were to type "/jtest #test", it would first join the channel #test, and then change the channel topic to "just testing." Another example along the same lines:

```
/love /say I love $1 ! | /say I really love $1 !!
```

If you were to type "/love everybody", mIRC would know to "/say I love everybody !" and then, "/say I really love everybody !!"

### **Example: Mode Changes (Opping, etc.)**

To put your often-used mode changes in the aliases section, you first need to know the normal structure of the mode commands. Opping somebody on IRC normally requires the command:

```
/mode #<channelname> +ooo <nick1> <nick2> <nick3>
```

...in which up to 3 persons can be mentioned. This can be nicely aliased with:

```
/op /mode # +ooo $1 $2 $3
```

To op John, Jake, and Joan, now you can just type: "/op John Jake Joan"

The # in this alias shows the use of one of other the pre-defined strings (# and \$!). The # always stands for the #channelname of the channel on which the alias is used... In the same way you can make aliases like...

```
/deop /mode # -ooo $1 $2 $3  
/ban /mode # +b $1  
/kick /kick # $1  
/dbk /mode # -o $1 | /kick # $1 | /mode # +b $1
```

### **IMPORTANT: Proper spacing!**

This is very important! You MUST include spaces between strings and the rest of your text! For example:

```
/like /say I really like $1!!!
```

...will NOT work! There needs to be a space between the string "\$1" and your "!!!"s. (This can sometimes make your text look awkward when using commands like /say, so be careful with your phrasing.)

```
/like /say I really like $1 !!!
```

...WILL work just fine.

This is true of ALL strings. To make one particular case easier, specifying channels, a few special strings have been created that automatically prefix the parameter with the # character. These are # \$1, # \$ \$1, and # \$? . See below for details.

In SOME cases you CAN do: /like /say I really like \$1's topic, but keep in mind: NOT ALWAYS !! You have to test this in every case you want to use it...

### **Where can I use Aliases?**

Aliases can be used in the edit box (the line or lines you normally use to type commands in), and you can now even use aliases in the Popups and the Remote sections that will be explained in the following sections. The function key aliases can also be used on selected nicks in the channel names-list and in query windows, etc....depending on your alias definitions.

### **String Definitions.**

By now, you should have a decent grasp of the Aliases section. I'll conclude this section with a full list of the strings that you can use in your aliases. At this point, make sure to try all of the examples given until you fully understand how to do it yourselves!

# A pre-defined string that always contains the channel name where the alias was executed. This is important because many commands, like the /mode commands, require that you specify a channel name.

Example:

```
/op /mode # +o $1
```

Type "/op Cindy", and it will op the person with the nick Cindy in whatever channel you type this.

\$1 The simplest string. It contains one word, the first word after the alias command. By the same token, \$2 would indicate the second word, \$3 the third, etc.

\*1 This string can contain a LOT of words... It will contain the first word, \*and\* everything after the first word.

Try:

```
/tell /say I'd like to tell all of you something - *1
```

The command "/tell mIRC is really great" with this alias will result in the text: "I'd like to tell all of you something - mIRC is really great". \*2 would contain the second word and everything after it. \*3 would contain the third word and everything after it. etc. Also try...

```
/note /notice $1 *2
```

...and type "/note <nick> <whatever text>"

And...

```
/mess /msg $1 *2
```

...and type "/mess <nick> <whatever text>"



(Make sure you realize that these commands would have given the same result if you defined the aliases as '/note /notice \*1' and '/mess /msg \*1' respectively.) Note : you can also use constructions like \*1-4

**\$\$1** An alias containing this kind of string will be executed ONLY if a parameter is given; otherwise, the call to the alias will be ignored.

For example, if you try this alias:

```
/j /join $$1
```

and then type, "/j" (instead of "/j #channelname")... mIRC will ignore you because you didn't include a channel name. Otherwise, it works the same as \$1 -- it contains the first word. \$\$2 would contain the second word, \$\$3 the third, etc.

**\$?** This is a nice one!

An alias with this string in it can be started without specifying the parameter. mIRC will display a Parameter Entry dialog box, in which text, a nick, or whatever, can be entered. Try it with these examples:

```
/j /join $?"  
/hug /me hugs $? firmly
```

**\$?="text"** An "extra special" version of the \$? string. mIRC will again display the Parameter Entry dialog box, but will now ask for the specific kind of information to enter. Try these:

```
/j /join $?="Which channel to join?"  
/hug /me hugs $?="Who to hug?" firmly
```

**#\$1 -and- #\$\$1 -and- # \$?** These special versions of the \$1, \$\$1, and \$? strings are used in connection with channels only. All these do are affix the # character to the beginning of the parameter.

Try...

```
/j /join # $1"
```

You'll be able to type "/j test" instead of "/join #test".

Also try:

```
/j /join # $?="Give channel name to join (No #)"
```

**\$me** This pre-defined string always contains your current nick! You can use it in all kinds of situations.

For example:

```
/nice /me thinks $me is a nice nick !  
/offer /me offers mIRC version 3.7 Do "/ctcp $me xdcc send #1" to get it.  
/away /me is away ( *1 ) | /away *1 | /say Messages to $me will be stored.
```

Or give the alias:

```
/nick /say I like $1 better than $me | /nick $1
```

a try, and change your nick then by "/nick <newnick>"

\$! This pre-defined string is used only in conjunction with \$?. If you've used \$? earlier in the alias, you can use \$! Later on to recall whatever parameter was typed in.  
For example:

```
/j /join $?="Which channel to join?" | /say I like channel $!
```

Type "/j" and it will display a Parameter Entry dialog box asking you to type in the name of the channel to join. If you type "#test", it will join #test, and then say into the channel: "I like channel #test".

Also try:

```
/friend /say $? is my friend | /say I like $! a lot
```

\$read [-l#] <filename.txt> This predefined string is a funny one. It will pick a random line from the specified file and insert that line into the alias.... The line can contain plain text and, also, commands !! (to make it easier ;o) Also \$!read which is the same as \$read except that it isn't evaluated the first time it is passed through the alias routines. This allows you to use it in a /timer command if you want the \$read to be re-evaluated each time.

**In the file you get the random lines from, the FIRST line should state the total of text lines in the file ! Else always only the first line will be used .... not very random, huh ?**

For example:

```
/fun /say $read c:.txt  
/kick /kick # $1 $read c:.txt  
/silly /say $read -l5 silly.txt
```

\$url This predefined string always contains the current page your www browser Netscape shows you...

For example:

```
/show /say Hey guys, I found a superb www-page ! Check out $url  
/mu /me found this great URL : $url
```

More predefined strings :: \$ip Your IP Address \$time The time on your PC \$date The date (european order) (YES !! mIRC is -not- made in the USA ;o) \$server The server you use \$snicks All nicknames you have selected in the channels names list.

### **Special Alias Definitions ; The Function keys.**

A special alias definition is possible which can launch commands using the Function keys. Define some aliases as follows:

```
/f1 /say Hello to you all  
/f2 /topic # This is a new topic
```

You now can type "/f1" to say hello to all on a channel ... but also pressing the "F1" function will work !! Other examples:

```
/f5 /me will happily send you the new mIRC.  
/f6 /join #mirc  
/f9 /dcc send $1 c:.zip  
/f10 /dcc send $1 c:.zip
```

```
/f11 /dcc send $1
```

The F9 key will send the mIRC FAQ to a person selected from the channel names list. Don't forget to select a nick first ! If you don't, you will see strange (but logical) things... You can even use the F9 key in a Private, Query or DCC Chat window with this alias definition ! The F11 key will prompt you for the nick to whom to send a file... Well.... invent your own custom use of the F keys ..... have fun ! You can also use -shift- and -ctrl- in combination with the F keys. Just define aliases like :

```
/f1 /say I used F1 to say this  
/sf1 /say I used Shift+F1 to say this  
/cf1 /say I used CTRL+F1 to say this
```

Note : Almost all aliases I could think of are given in section 8 ! Check them out.

[Table of contents](#)

## 7-2 The Popups section.

Popups are the menus that appear when you right-click with your mouse. If you haven't found this nice feature of mIRC already, give it a try NOW! Some popup menu's are predefined in the mIRC.ini from the mirc.zip-file, but they are fully customizable under Tools/Popups. If your copy of mIRC doesn't seem to be equipped with some prefab popup menu's (the Tools/Popups section is empty then), it is almost certain that you didn't make sure that the prefab popups.ini file was correctly copied from the mirc.zip to the mirc directory, or the path to where this prefab popups.ini can be found is not correctly specified.

The popup commands work almost exactly like aliases. All of the strings mentioned in Section 7-1 are allowed in popups (and vital for the proper use of them). If you haven't read the Aliases section (7-1), please do so now! You need to know that information before moving on.

Also: Definitely read the Help file's explanation of popups, as well as take a look at the many popup examples included in mIRC under Tools/Popups. The more examples you try, examine, and mess around with, the better your grasp of mIRC will be.

### **Example: /join**

Let's say that you join a particular channel frequently, #test. Rather than have to type in "/join #test", wouldn't it be easier to just right-click and select a menu item to join you there automatically? The following popup would do that. Just enter it into Tools/Popups, making sure that "Main Window" is selected at the top. (Not Private or Query window, nor Status window!)

```
Join the Test Channel:/join #test
```

Now right-click over your Status window or any channel window and select "Join the Test Channel". mIRC will do exactly that.

### **Example: /away**

Here's an example of "away" and "back" popups... you type in the reason you're away, and it's sent to all the channels you're on. (That's what the /ame command does, it sends a /me to every channel you're on.)

```
Set Away...:/ame is AWAY ( $?="Give a reason" ) | /away Please try later ( $! ) !!
Set Back:/ame is BACK | /away
```

(Note again that on some systems you may need to scroll to the right to see the full text of some examples.) Note that the "Set Away" example uses multiple commands.

## Heirarchal Menus

If you were to add a few popups like that, your screen would quickly get cluttered every time you right-click. So you can organize your popups into heirarchal menus. Try entering all of the following popups:

```
Say Things
.Hello:/say Hello folks
.Goodbye:/say Goodbye folks
.Hug Somebody:/me gives $? a big hug
```

Here we're specifying that the last three commands should show up under "Say Things" by: 1) including them after "Say Things", in the order we want them to appear, and 2) prefacing them all with "." <-- a period. When you right-click, you'll see the "Say Things" menu item. When you select that, you'll see "Hello", "Goodbye", and "Hug Somebody" menu items. Select from those to perform one of the actions specified above. If you want to get really complicated, you can make your popups up to four levels deep. For example, try:

```
Comments About People
.Compliments
..Nice Today:/say Wow, $? -- you're being very nice today!
..Helpful:/say Thanks for being so helpful, $?
..Friend:/say I'm glad to have $? for a friend.
.Insults
..Jerk:/say Leave me alone, jerk!
..Idiot:/me wonders why $? is such an idiot sometimes.
..Enemy:/say I'm proud to have $? as my enemy!
```

And so on! Just order them properly, using the periods. (If the manner in which the above is organized doesn't "click" with you, definitely try it out! Enter the above "comment" section into Tools/Popups. It's a lot easier to grasp when you see it working.) All popup menu items can be put under a Function key too! That's handy as a replacement for often-used popups.... (Make sure to use the correct syntax and put them (ie.the function key replacements) in the alias section)

### Table of contents

## **7-3 The Remote Section.**

Under Tools/Remote/... the Remote section will be found. Clearly the most advanced part of mIRC, you can do things with mIRC's Remote section that would normally require scripting or special bot software...

Things like: offering files (an XDCC list); responding to specific channel or private message text; customizing CTCP responses; setting up various user levels for your friends; ...almost anything that happens on IRC can be automatically responded to, and your response can make use of almost any mIRC command or commands.

The Remote syntax can be a bit complicated at times, but it's a simple extension of the formats explained previously for Aliases and Popups. If you understand those well, including the Strings, then please proceed. If not, go back and read (or re-read) Sections 7-1 and 7-2. The information presented there is ABSOLUTELY REQUIRED LEARNING before attempting Remote.

It's also important to remember why Remote is called Remote. The Remote section is designed to respond to things done by users OTHER than yourself -- in effect, it allows others to "remotely control" your mIRC session, according to the specific Remote features you implement. Many examples can't be tested by simply sending the commands or typing the text yourself. They have to be sent by some other client.

Remote consists of three parts: Remote Users (the user list), Remote Commands, and Remote Events.

[Table of contents](#)

### **7-3-1 Remote Users.**

See Tools/Remote, the box at the top left labeled "Users"...

In this part of Remote you can assign various user levels. You can assign certain levels to your friends and certain levels to your "enemies" (you can create a "shit list"), and a host of levels in between. Why is this important? Let's say that you've written a Remote Event designed to kick anyone who says the word "nonsense" in your channel (just replace this in your head with whatever foul word you'd like). We'll explain how to do this later (in Section 7-3-3), but for now it's just the idea that's important. Sounds reasonable, right? OK, but perhaps you'd ONLY like regular users to be kicked for saying "nonsense." You'd like some of your friends only to be warned. And you'd like to ignore it from still others, perhaps your closest friends. And if you don't care for someone much at all, perhaps you'd like to kick AND ban them for saying "nonsense"! To do these kinds of things (give different responses to different people), you need to setup a users list. For each user, you'll specify their nick or address and the user level you've decided to assign them. Here's an example of a valid user list:

```
1:nick1
1:nick2
2:nick3!account3@machine.subnet.net
2:*!account4@machine.subnet.com
3:*!*@machine.subnet.edu
4:*!*@*subnet.edu
5:*!account@*.subnet.edu
10:yournick!youraccount@yourmachine.net
```

The numbers you pick for the user levels are completely up to you. You can decide what each of the user level numbers can and cannot do, as you'll see in the next two sections. By default, mIRC assigns everyone not specified in the user list to level 1. (Depending on the setting in Tools/Remote/'Default level'. If you specified different levels to different users you of course also need commands and events that have different reactions or power to users of these levels... That will be explained later...

As you can see, either a nick or an address are permissible. Wildcards are allowed. There are some cases where you might want to specify your users twice, by both nick and address. This case (ON OP/ON DEOP/ON SERVEROP/ON NOTIFY) is explained in Section 7-3-3. You don't have to do all the editing by hand ... you could also use mIRC's edit box or command

line commands /auser, /guser and /ruser.

/auser <level> <nick or address> 'Auser' (Add user) adds whatever nick or address you specify to the users list at the level you specify. It does not check for a valid address. The nick or address is added exactly as you stated it.

/guser <level> <nick> [type] 'Guser' (Get user) adds a nick to the users list specified by address. To accomplish this mIRC does a /whois on the nick you specify, and adds the returned address on the specified level to the users list. This means that the specified nick has to be on IRC to be able to use the /guser command to add him/her to the list. By specifying the 'type' mIRC can add a user with several types of wildcarded addresses. This is analogous to the 'type' parameter used in the /ban command. Type ranges from 0 to 4.

/ruser <nick or nick! or address [type]> 'Ruser' (Remove user) removes a nick or address from the Users list. You must know exactly how the nick or address is stated in the User list for this command to work. If somebody is listed like nick!blabla@whatever.net you can also do "/ruser nick!". Mind the "!" to remove the entire line.

In mIRC version 3.5 an improved /ruser command is made so that it can work like /ban and /guser. /ruser <nick> [type] if you do NOT specify a type, then it works as usual and removes the specified nick from the user list. If you DO specify a type, it looks up the users address and removes that address from the user list.

Note: Using different address types allows you to specify a person or group of people with more or less wildcards. I'll show you the different results .....

```
No type mentioned; Kreet!*vonck@Wit399402.student.utwente.nl
type 0;Kreet!*vonck@Wit399402.student.utwente.nl
type 1;!*vonck@Wit399402.student.utwente.nl
type 2;!*@Wit399402.student.utwente.nl
type 3;!*vonck@*.student.utwente.nl
type 4;!*@*.student.utwente.nl
```

mIRC now keeps an internal database of the people on the channels you are on. This database is used to speed up the performance of commands like /ban, /guser and /ruser a whole lot. These commands normally will make mIRC do a /whois on the person you issue the command on. The internal list stores info like nick!user@address for all users that are on the same \*channels\* as you. Whenever a /guser, /ruser, /ban, /finger, /ignore or /dns command is issued, the internal list is searched first to find the nick's address. If no match is found, the normal /whois stuff is done. This will speed up your bans a lot...especially when you're lagged... If you are on a slow PC and/or you are on a lot of large channels, mIRC's updating of the database might slow you down quite a bit ... then its probably good to switch it off ! In the Tools/Remote dialog you can deselect the use of the 'Internal Address list'.

[Table of contents](#)

## **7-3-2 Remote Commands.**

See Tools/Remote, the box at the bottom, with "Commands" selected...

Remote Commands refers to a very specific type of command: CTCP commands. The Remote Commands section of mIRC does one thing and one thing only -- it allows you to make mIRC to respond to CTCP commands sent by other users the way -you- wish. There are a lot of ctcp commands defined according to the IRC RFC's (standards). The usual ones are version, ping, time, userinfo, clientinfo and finger. Every IRC client will respond in a standard

way if one of these commands is send to it. These ctcp commands are sent by or to you in the format:

```
/ctcp <yournick> <command with parameter(s) if any>
```

Consider the simple example of a ping. The other user would type:

```
/ctcp <yournick> ping
```

(You can also send yourselves a ping command! Try it! Type "/ctcp <yournick> ping".

Under normal circumstances, mIRC would send a standard ping reply [ctcp hisnick pong]. But with Remote Commands, you can redefine the program's response to the ping. It can now do almost whatever you'd like it to do when it receives a ping. On top of that, it can also send a normal ping reply. Or not. It's up to you.

There is ONE exception to this ... you cant hide your mIRC version reply.... We like the advertisement, you know....

You can also create new CTCP commands, and responses for them. This is how features like file offering (XDCC LISTS and SENDs) are handled. But right now, let's get back to our ping example.

### **Example: Custom Ping Response**

Place the following example into your Remote Commands. Go to Tools/Remote, make sure that "Commands" is selected, and "Listening" is checked, (When "Listening" is not checked, Remote Commands are ignored.) and write in the commands box:

```
1:ping:/notice $nick ping? Hmmm! ...pang pang pang!!
```

Now if someone pings you, your reply will be "ping? Hmmm ! ... pang pang pang !!" in a notice sent to the person who pinged you, and not the normal reply being 'pong'. Test this by giving the command "/ctcp <yournick> ping"!

If you want to be a bit more polite, you can add the redirection command to the end. :> It looks like a "smiley" but it's not.

```
1:ping:/notice $nick ping? Hmmm! ...pang pang pang!! :>
```

This will do the same thing as before, except it will also send your normal ping reply. Again, test it and see the difference! Assuming you actually tried the two given ping replies, and that you got them to work, let's go over the syntax used here a bit, and then we'll go back to the above example and why it does what it does.

### **Remote Command Syntax.**

Remote Commands, except for some special cases, are made up of 3 portions. Each of the portions are seperated by a colon. The general syntax looks like:

```
<command_level>:<ctcp_command>:<triggered_command(s) with parameter(s)>
```

Lets go over the separate portions one by one:

<command\_level> This command will respond to users with a user level equal to the command level, and to users with higher levels, unless other commands are created for them.

<ctcp\_command> The CTCP command can be named anything you want it to be. It can be a standard, already existing CTCP command, like Ping or Version, or it can be one of your own creation. (like OPME, DIE, LEAVE, XDCC LIST,.....)

<triggered\_command(s) with parameter(s)> In this part you define the (to be) triggered command(s). Their format and the use of multiple commands is almost exactly as you define them in the aliases and popups. The strings in this section are filled in by mIRC automatically... You only have to make sure that the strings you use are defined at all... ie. make sense if used in the remote commands ... (For instance the \$chan is NOT defined in the remote commands !!)

Let's take a look at our ping example and try to dissect it.

```
1:ping:/notice $nick ping? Hmmm! ...pang pang pang!! :>
```

1 is the command level. All users with a user level equal or higher than 1 can execute or use this command.

Ping is the CTCP command this Remote Command is going to react to.

The triggered command is a /notice. In this case a notice send to \$nick containing the entire line "ping? Hmmm! ... pang pang pang!!" \$nick is a pre-defined parameter that contains the nick of the person who gave you the CTCP command ... in this case he pinged you. (More about the pre-defined parameters soon.)

Summarising, the effect of the command "/ctcp <yournick> ping" is a "/notice <nick> ping? Hmmm! ...Pang pang pang!!" This is executed as soon a CTCP Ping is received.

The redirection used (the :> part) may seem hard to understand at first. Normally, this would be used to redirect the command to another command. For example, placing ":> version" there would cause it to immediately go to your response for the CTCP Version command. But because we've left it empty (just :>), and because Ping is a command that mIRC has a default response to (like with Version, Time, Userinfo, Clientinfo, etc.), the default response is given -- a standard ping reply -- on top of your addition.

Take a look at this example of how redirection would usually be used:

```
1:help:/notice $nick Sorry I can't help you... :> morehelp
1:morehelp:/notice $nick But.... join #mirc to find it all out ??
```

The command "/ctcp <yournick> help" will result in 2 notice lines sent to the remote user...

-yournick- Sorry I can't help you...

-yournick- But.... join #mirc to find it all out ??

...because mIRC responds to the "help" command, and then redirects it to "morehelp."

Rather than using redirection, you could also use multiple commands, utilizing the | character, as with Aliases and Popups. That way a single Remote Command could've sent both notice lines, rather than using redirection. The use of the | is very straight forward, and we'll cover it later. Just glue two remote commands together using the | (pipe).

## **Different reactions to people with different user levels**

You can fine tune your reactions to remote commands by assigning different levels to them. mIRC will always respond to a user with the highest leveled reply that fits to the command he/she gave.. Lets analyse it with these example lines...



```
1:ping:/notice $nick ping? Hmmm your levels are tooooo low...
2:ping:/notice $nick ping? Hmmm! ...pang pang .... Quite dead !! :>
3:ping:/notice $nick ping? ...pang !! Hmmm... missed... :>
6:ping:/notice $nick ping? Hmmm! ...Well I like you .. you're allowed to ping me.. :>
```

A level 1 user won't get a ping reply but a nice "ping? Hmmm your levels are tooooo low..."  
A level 2 user (2:nick3!account3@machine.subnet.net) will not survive it when he does a /ctcp <yournick> ping ;o) He will see "ping? Hmmm! ...pang pang .... Quite dead !!"  
A level 3 user (3:\*!\*@machine.subnet.edu) will get " ping? ...pang !! Hmmm... missed..."  
To level 4 and 5 users no specific reaction is defined ... The highest to them available command is the level 3 one... Therefore these people (5:\*!account@\*.subnet.edu) will see the level 3 reaction too !!  
Level 6 and higher users will all get the level 6 ping reply...

This leveling can be done for ALL remote commands (and events) .... Make sure you get the point here !!! Maybe re-read the above sections ??

### The Pre-defined Strings

\$nick, \$me, \$address, \$parms, \$parm1, \$parm2, \$parm1\*, \$parm2\* etc. are all pre-defined strings. You can use them wherever you want in the lines triggered by remote commands...

\$nick           The nick of the person who sent the command.  
\$me             Your nick.  
\$address        The address of the person who sent the command.  
\$parms           Any text the person might've sent FOLLOWING the command.  
                  If they'd sent: '/ctcp <yournick> ping hello there' ...we would pick up "hello there" in \$parms if the by you defined commands is reacting to 'ping'.  
\$parm1, etc.    Lets you break down \$parms into individual words.  
                  In the above example, \$parm1 would contain "hello" and \$parm2 would contain "there". Etc.  
\$parm1\*         This one is a bit odd.... It contains the first word and all after it.  
                  Analogue \$parm2\* contains the second word and all after it....  
\$read <textfile.txt> This parameter will pick a line from the specified file randomly.  
                  Read in the mIRC help for more info on \$read.  
\$url            Your current Netscape www browser page

The \$chan is NOT DEFINED in the Remote Commands ... \$chan normally contains the channel from which a commands is given ... CTCP commands dont have to be given to you from a channel ... or the person sending the command is on more channels ... Get the point ??

### Example: Here's an example using the pre-defined \$address parameter...

```
1:address:/notice $nick Your address is $address
```

If a user were to send a '/ctcp <yournick> address' command to you, you would reply with a notice reading, "Your address is <whatever his or her address is>".

### Example: Opping friends on demand. (Using \$parms)

If you're not clear on the basics of how all of this works yet, don't worry. Try to follow along with the examples and see if it "clicks" in your brain. Re-read the above examples if necessary. But don't attempt to write your own Remote Commands or move on to the next section until it "clicks."

```
1:opme:/notice $nick Sorry, you won't get opped on $parms .. your level is too low
5:opme:/mode $parms +o $nick
```

If a level 5 friend sends '/ctcp <yournick> opme <#channelname>', this command will op them in the specified channel. Level 1 people WON'T get opped whatever they try.... Here another predefined parameter is used -- \$parms, which refers to everything the user typed after the CTCP command (opme). In this case, it's the channel name. The \$parms can be used to process long lines given by a remote user in a CTCP command...

```
5:return:/notice $nick $parms
```

### **Multiple Commands.**

A Remote CTCP command can also trigger multiple commands....

```
5:opme:/mode $parms +o $nick | /notice $nick You are opped on $parms
```

Then your level 5 friend sends the command:

```
'/ctcp <yournick> opme #channelname' What happens??
```

In effect two commands are triggered by the Remote CTCP opme command...

```
1st  "/mode #name +o friend"
2nd  "/notice friend You are opped on #name"
```

Another CTCP command triggering multiple commands:

```
10:part:/part $parms | /notice $nick I have left channel $parms
```

And if a level 10 user sends you:

```
'/ctcp <yournick> part #channelname'
```

This command makes you leave a channel on request, and tells the requester you did so... \$parms contains the channel name you are asked to leave. \$nick is the nick of the person who asked.

### **Other Examples.**

```
1:ping:/notice $nick Please don't ping me.
```

...would respond with: "/notice <nick> Please don't ping me."

```
5:time:/notice $nick Time to get a watch.
```

...would respond with: "/notice <nick> Time to get a watch."

```
10:whoareyou:/notice $nick I am who I say I am :> whoaretwo
10:whoaretwo:/notice $nick And I say that I'm YOU!
```

...would respond with: "/notice <nick> I am who I say I am" followed by

"/notice <nick> And I say that I'm YOU!"

### Examples of Remote Commands that WILL NOT WORK

```
1:zap:/notice *1:$nick No zaps allowed! :>
```

**\*WILL NOT WORK\***

...Why? You're using an empty redirection command (:>) on a CTCP command that you made up yourself. This only works on the standard CTCP commands like Ping, Version, Time, etc.

```
10:version:/notice $1 $2 | /notice $1 *3:$nick Version? I ain't got no stinkin' version.
```

**\*WILL NOT WORK\***

...Why? You're using the OLD Remote syntax !!!! (You dummie ...)

```
10:version:/notice *1:$nick Version? | /notice *1:$nick I ain't got no stinkin' version.
```

**\*WILL NOT WORK\***

...Why? You're using the OLD Remote syntax mixed with the new one.... (Hihihi ...)

[Table of contents](#)

### 7-3-3 The Remote Events section.

See Tools/Remote, the box at the bottom, with "Events" selected...

As you've seen, Remote Commands only deal with CTCP commands. Remote Events deal with just about everything else that can happen on IRC. If that sounds daunting, don't worry. As explained earlier, these lessons are cumulative. If you understand Remote Commands, Remote Events will be a piece of cake. You've already learned the really difficult stuff.

Warning: In general mIRC is very sensitive to poor Remote Event syntax. Using incorrect syntax can cause a GPF (General Protection Fault), so be careful! If ever you suddenly experience stability problems with mIRC, this is the part to check first!

#### Remote Event Syntax.

The Remote Events are: ON TEXT, ON JOIN, ON PART, ON KICK, ON OP, ON DEOP, ON INVITE, ON NICK, ON QUIT, ON TOPIC, ON SERVEROP, ON NOTIFY, ON MODE, ON SERVERMODE and ON SMODE. Besides ON TEXT the similar ON ACTION, ON NOTICE, ON SNOTICE, ON CHAT and ON SERV are available for more specific text origins.

Exactly what they do, and how to use them, will be explained shortly. First, some quick examples and an explanation of the syntax used.

The syntax is essentially the same as with Remote Commands. Here's a sample Remote Event for ON TEXT:

```
1:ON TEXT:hello:#:/msg $nick Hello back!
```

Place that example into your Remote Events. Go to Tools/Remote, make sure that "Events" is selected (not commands), and "Listening" is checked. (When "Listening" is not checked, Remote Events and Commands are ignored.) This Remote Event is triggered when the word "hello" is found in any active channel. mIRC's response is to send a private message to the

person who wrote it, saying "Hello back."

Here's another example, this time using ON JOIN:

```
1:ON JOIN:#:/notice $nick You are very welcome to $chan
```

This one is triggered whenever a user joins a channel you're on. A notice is sent to them that says, "You are very welcome to <channelname>".

If you've made it this far, and understood Remote Commands, this should seem fairly simple to you.

Remote Events have either 3, 4 or 5 portions divided by colons. All of the events have 4 portions, except for ON TEXT which has 6, and ON QUIT and ON NICK have only 3. All of the events \*except\* for ON TEXT, ON QUIT and ON NICK are formatted like so:  
<event\_level>:<event>:<where>:<triggered\_command(s) with parameter(s)>

<event\_level> This event will effect users with this user level, and any higher levels. Unless other event entries are created for them. Just like command\_level in Remote Commands.  
<event> The name of the event we're listening for. ON JOIN, ON OP, etc.  
<where> #, #name.  
Where are we listening for this event? Either # (any channel) or the name of a specific channel (like #test). EXCEPTION: Not used with ON NICK or ON QUIT. Then just skip this part.  
<triggered\_command(s) with parameter(s)> The (to be) triggered command. Just like in Remote Commands except that we have a few different pre-defined parameters.

...and of course, the big exceptions here are ON TEXT and ON ACTION which work like so:  
<event\_level>:<event>:<what\_text>:<where>:<triggered\_command(s) with parameter(s)>

<command\_level> Same as above.  
<event> Well, obviously, it's going to be ON TEXT or ON ACTION  
<what\_text> Specify the word or phrase to listen for. Like "hello" in the first example.  
<where> #, #name, ?, \*. Just like above, you can tell ON TEXT and ON ACTION to listen on all channels (#) or on a specific channel (like #test). But you can also tell it to listen to all of the private messages sent to you (?) or to both channel and private message text .

An ON TEXT or ON ACTION event can't listen for things said in DCC Chats or Server messages.

<triggered\_command(s) with parameter(s)> Same as above.

## The Events

Here are the various Remote Events that mIRC can respond to...

\* ON TEXT Occurs when specified words/phrases are found in a channel or private message.  
ON ACTION works just the same ...  
Example:

1:ON TEXT:nonsense:#:/kick \$chan \$nick No nonsense!

This will look for the word "nonsense" in any channel and kick the user who said it.

Example:

1:ON TEXT:nonsense:#test:/kick \$chan \$nick | /notice \$nick Hello \$nick , you said nonsense' and that's not allowed on #test...

If the word "nonsense" is said in #test, the sayer is kicked (if you're 'op' on #test) and sent a notice: "Hello <nick> , you said 'nonsense' and that's not allowed on #test..."

Example:

5:ON TEXT:help:?:/notice \$nick I wish I could help you, but I can't. If it finds the word "help" in any of your private messages, it replies to the user, "I wish I could help you, but I can't" in a notice.

You can specify wildcards in the text mIRC should react to:

- \* reacts to any text
- word if a user said only (and exactly) this word the event will trigger. Exact match required.
- word\* if a user started a line with this word the event will trigger.
- \*word if a user ended any line with this word the event will trigger.
- \*word\* if a user said this word or phrase anywhere the event will trigger.

Example:

5:ON TEXT:help:?:/notice \$nick I wish I could help you, but I can't.

This event will only react to a call for help if the msg only contained the word help ... not if a line containing the word 'help' was sent to you...

Example:

5:ON TEXT:hello\*:?:/notice \$nick Hello to you too !

This event will only react to private message lines sent to you starting with the word 'hello'.

\* ON ACTION See ON TEXT .... it works exactly the same....

\* ON NOTICE See ON TEXT .... it works exactly the same....

\* ON CHAT Works like ON TEXT but listens to a DCC Chat text.

Example:

1:ON CHAT:blabla:/msg \$me \$nick said \$parms to you in a DCC Chat.

\* ON SERV Works like ON CHAT but listens to DCC Fileserver text.

Example:

1:ON SERV:get mirc:/echo server 6 To \$nick the mIRC zip is sent.

\* ON JOIN Occurs when a user joins a channel.

Example:

1:ON JOIN:#test:/notice \$nick Welcome to the #test channel!

Sends a notice saying "Welcome to the #test channel!" to anyone who joins #test.

Example:

```
25:ON JOIN:#cool:/kick $chan $nick You're not welcome here!
```

Kicks any level 25 user who joins #cool with the message, "You're not welcome here!"

\* ON PART Occurs when a user leaves a channel.

Example:

```
10:ON PART:#:/notice $nick Hey, thanks for stopping by $chan !
```

Sends a notice saying, "Hey, thanks for stopping by <channel> !" to any level 10 who leaves a channel you're on.

Example:

```
90:ON PART:#:/msg $chan Yahoo! $nick is gone!
```

Sends a message to the channel saying, "Yahoo, <nick> is gone!" if any level 90 leaves a channel you're on.

\* ON KICK Occurs when a user is kicked from a channel.

Example:

```
100:ON KICK:#:/kick $chan $nick | /invite $knick $chan | /notice $nick That person is my friend!
```

If any level 100 person is kicked from any channel, you invite them back and send "That person is my friend!" to the kicker. NOTE: \$knick is the parameter for the kicked nick in this event. Also note that this example shows multiple commands. It works just like in Remote Commands.

\* ON OP Occurs when a user is opped.

Note: This event requires people to be mentioned by NICK in your Remote/Users list !

Example:

```
1:ON OP:#test:/notice $opnick You were opped by $nick . Welcome!
```

If someone is opped on #test, they're sent a notice: "You were opped by <opper>. Welcome!" NOTE: \$opnick is the parameter for the opped user's nick in this event.

\* ON DEOP Occurs when a user is deopped.

Note: This event requires people to be mentioned by NICK in your Remote/Users list !

Example:

```
1:ON DEOP:#:/msg $opnick Tough luck. You were deopped by $nick
```

If someone is deopped, they're sent a message: "Tough luck. You were deopped by <deopper>. NOTE: \$opnick is the parameter for the deopped user's nick in this event.

Explaining the ON OP, ON DEOP, ON SERVEROP as well as the ON NOTIFY event an important remark should be made. Due to the fact that servers only report the nick's of the people who are opped, deopped or notified on a channel or IRC, the ON OP, ON DEOP, ON

SERVEROP and ON NOTIFY events only work properly if the nicks of the people who are opped, de-opped or notified are mentioned in the Users list. Specifying even the full address (nick!account@machine.address.net) is NOT enough... mIRC won't distillate the nick from such a statement. The only exception to this is if you wish the event to affect everyone by giving it an access level of 1, or whatever you have set the default user level to. This is the only case where it is not necessary to have a user in your User list by nick.

If you want to give people a level fitting to your ON OP, ON DEOP, ON SERVEROP or ON NOTIFY events you have to put them in your Users list by nick, like:

```
3:friendasnick
5:friendbsnick
5:friendcsnick
10:friendbsnick!accountb@bsmachine.net
```

This construction can give you a good reason to specify people in your Users list by both their full address (wildcards are permitted) and their nick. This could then guarantee the proper reaction to an ON OP ON DEOP or ON SERVEROP without giving ALL people around with by accident (?) the same nick the same rights as your friends (or enemies).

\* ON INVITE                Occurs when you are invited to a channel.

Example:

```
5:ON INVITE:#test:/join $chan | /describe $chan thanks you for inviting him!
```

If invited to #test, join the channel and use an action to say "<me> thanks you for inviting him!"

\* ON NICK                Occurs when a user changes his or her nick.

Example:

```
1:ON NICK:/notice $newnick I thought $nick was a nicer nickname!
```

Send a notice to anyone who changes nicks, "I though <your old nick> was a nicer nickname!"

NOTE: \$nick is the parameter for the old nick, and \$newnick for the (guess what?) new nick.

NOTE: The <where> portion is not used in ON NICK.

\* ON QUIT                Occurs when a user quits IRC.

Example:

```
2:ON QUIT:/notice $me Time to party! $nick just quit the IRC!
```

Whenever someone quits, send a notice to yourself: "Time to party! <Nick> just quit the IRC!"

NOTE: The <where> portion is not used in ON QUIT.

\* ON TOPIC               Occurs when a channel's topic is changed.

Example:

```
1:ON TOPIC:#:/msg $chan Wow, I love the new topic!
```

Whenever a channel topic changes, sends a message to the channel saying, "Wow, I love the new topic!"

\* ON SERVEROP Occurs when a server gives operator status (back) to a user on a channel.

Note: This event requires people to be mentioned by NICK in your Remote/Users list !

Example:

```
1:ON SERVEROP:#mirc:/mode $chan -o $nick | /notice $opnick Sorry, serverops not allowed on $chan
```

Whenever a person is opped on #mirc by a server, the opped person is deopped and receives a notice "Sorry, serverops not allowed on #mirc" Serverops normally occur after a netjoin when the two splitted nets exchange and update all current channel modes.

\* ON NOTIFY This event triggers as soon as people in your notify list join IRC.

Note: This event requires people to be mentioned by NICK in your Remote/Users list !

You could use this event to do a /whois on the notified nick to make sure it is the correct guy.... Remember that this event ONLY works on nicks mentioned in the remote/user list by NICK.... (like ON OP, ON DEOP and ON SERVEROP) This is due to the IRC protocols which don't pass through the address of the notified person...

Examples:

```
1:ON NOTIFY:/notice $nick If you are the real $nick then please join #mine !
3:ON NOTIFY:/whois $nick
5:ON NOTIFY:/beep 10 50 | /whois $nick
```

\* ON MODE

\* ON SERVERMODE These events can force a certain set of channel modes.

Use this event to make sure that when someone changes the channel modes, mIRC enforces the channel modes YOU want instead.

Syntax and examples :

```
1!:ON MODE:#name:-spmi+tnlk 20 mykey:/notice $nick illegal $chan mode change!
1!:ON SERVERMODE:#name:-spmi+tnlk 20 mykey:/notice $me illegal $chan mode change by server !
```

The ON MODE's react to any channel mode change, so there is no way to react to specific mode changes. This structure is certainly not perfect but it works...

Popular settings are :

```
1!:ON MODE:#mirc:-spmilk+tn:/notice $nick illegal $chan mode change!
1!:ON SERVERMODE:#mirc:-spmilk+tn:/notice $me illegal mode change!
```

Keep in mind that if you allow a higher level person to change mode settings without you reacting, lower level users can mess that up ! As soon as they change a mode (or the server does) then the forced modes are set; mIRC does NOT keep in mind that part of the new modes might be set by higher level persons ... Also beware of loops of more people on a channel force not matching modes !!

\* ON SNOTICE This event listens for server notices. Server notices tell you all kind of things happening at and with the IRC server you use. To receive them you have to set your user mode to +s (/mode #nickname +s). Most people don't need this setting... It is mostly ment for IRCops and server maintainers. With this event you can filter the server notices send to you and you can filter them ie. prevent them from being displayed or automatically react to them. The event syntax is very much like the ON TEXT event...



1:ON SNOTICE:servers\_text:/echo 6 status The server noticed you about \$parms

### Table of contents

## **7-3-4 Various flags you can use in front of Command or Event lines in Remote.**

Finally some remarks have to be made to the various flags you can use in the Remote section. (With Commands and/or Events)

### **The ! flag. (Events only)**

This flag makes that events are not triggered by things you do yourselves....

Example:

```
1!:ON JOIN:#mine:/notice $nick Welcome here.
```

### **The = flag. (Events only)**

This flag is used to prevent anything from happening.... you can use it to not bother higher level people with events meant for lower level people.

Example:

```
1:ON JOIN:#mine:/ctcp $nick VERSION
3:ON JOIN:#mine:=
```

Level 1 and 2 users will be versioned while level 3 and higher users will not see anything at all,... nor you. (In version 3.2 you had to set up an harmless event to do this.)

Example:

```
1:ON JOIN:#test1:/notice $nick Welcome to #test1
1:ON JOIN:#test2:/notice $nick Welcome to #test2
3:ON JOIN:=
```

Level 3 and higher users won't be noticed Welcome.... lower level users only on #test1 and #test2.

### **The + flag.**

This flag makes a certain command available ONLY to users with the EXACT level of the command... Higher (and of course lower) level users can't use/access it !

Example:

```
1:ON JOIN:#mine:/ctcp $nick VERSION
+3:ON JOIN:#mine:/notice $nick Welcome here.
```

All users EXCEPT level 3 users will be versioned on channel 'mine' ... Level 3 users will be welcomed...

Example:

```
+5:ON JOIN:#mirc:/msg $nick Welcome level 5 user!
```

### **The ; and REM flag.**

These flags can be used to switch off commands or events temporarily by 'commenting them out'.

Example:

```
;1:ON JOIN:#mine:/notice $nick Welcome here.  
REM 1:ON JOIN:#mine:/notice $nick Welcome here.
```

### **The \* flag.**

This flag makes mIRC to execute an event or command ONLY if you're operator on the channel the event or commands if used on.

```
*1:ON JOIN:#mine:/notice $nick Welcome here.
```

### **The me flag.**

This flag is made to make sure you can make events only working if somebody with exactly the same address as you use does something. This other person normally would be your 2nd instance of mIRC running from the same PC. Keep in mind that with proper leveling of your users in the Remote/users section you never (?) need this flag.

In mIRC version 3.7 some additional flags for the EVENTS are introduced. With these flags you can make mIRC to take into account the level of the person that started an EVENT. These flags can only be used with the ON OP, DEOP and KICK events since these are the only events that deal with somebody doing something to somebody else.

The three flags you can add are >, < and =. But you can make mathematical combinations of them leading to a set of 6 flags: <, >, <=, >=, <> and =.

These flags will make the event only be triggered if FIRST the level of the event fits the Opped, Deopped or Kicked guy (m/f) and then SECOND the level of the activator meets the expression <level-activator> [flag (mathematical expression)] <level-of-event>

You look like you need some examples ....

Assume you being the GUARD with a Friend at level 2 and an Activator who can have several different levels.... Also imagine these simple EVENTS :

```
<2:ON DEOP:#test123:/msg $chan the < triggered  
>=2:ON DEOP:#test123:/msg $chan the >= triggered
```

With the activator at level 1.

```
*** Activator sets mode: -o Friend  
<GUARD> the < triggered
```

Your Friend has level 2, the Activator has level 1,  $1 < 2$  is valid (one is smaller than 2), the first level 2 ON DEOP event is triggered...

With the activator at level 2.

```
*** Activator sets mode: -o Friend  
<GUARD> the >= triggered
```

Your Friend has level 2, the Activator has level 2,  $2 < 2$  is invalid,  $2 >= 2$  is valid (2 is larger than or equal to 2), the second level 2 ON DEOP event is triggered...

With the activator at level 3.

```
*** Activator sets mode: -o Friend
<GUARD> the >= triggered
```

Your Friend has level 2, the Activator has level 3,  $3 < 2$  is invalid,  $3 >= 2$  is valid (3 is larger than or equal to 2), the second level 2 ON DEOP event is triggered...

As you can see its pretty straight forward once you got the idea .... read this over till the coin drops... More examples :

```
<2:ON DEOP:#test123:/msg $chan the < triggered
>2:ON DEOP:#test123:/msg $chan the > triggered
```

With the activator at level 1.

```
*** Activator sets mode: -o Friend
<GUARD> the < triggered
```

With the activator at level 2.

```
*** Activator sets mode: -o Friend
```

With the activator at level 3.

```
*** Activator sets mode: -o Friend
<GUARD> the > triggered

=2:ON DEOP:#test123:/msg $chan the = triggered
```

With the activator at level 1.

```
*** Activator sets mode: -o Friend
```

With the activator at level 2.

```
*** Activator sets mode: -o Friend
<GUARD> the = triggered
```

With the activator at level 3.

```
*** Activator sets mode: -o Friend
```

[Table of contents](#)

## **7-3-5 Variables**

See Tools/Remote, select the box called "Variables" ...

mIRC now allows you to define all kinds of variables. Variables always start with a % and can have names of any length. The variables are stored between sessions in a variable file in the mIRC directory. Variables can contain numbers, words or even lines of text. On variables containing numbers you can apply mathematical manipulations to increase, decrease, add and subtract variables with given values or eachother.

Available commands are : (q=quiet; no status report of the operation is given)

```
/set [-q] <%var> [value] To create a variable
/unset [-q] <%var> [%var2] ... [%varN] To delete a variable
/unsetall To delete all variables
/inc [-q] <%var> [value] To increase a variable with a value (number or variable)
/dec [-q] <%var> [value] To decrease a variable by a value (number or variable)
```

With the aid of these commands you can make all kinds of handy aliases and remote lines!  
Let me give a bunch of examples :

Remote commands (see later) :

```
2:xdcc send #1:/dcc send $nick c:\temp\serve\mir40.zip | /inc %mir4 1
2:xdcc send #2:/dcc send $nick c:\temp\serve\mircfq24.zip | /inc %faq 1
1:stats:/notice $nick Sending stats: mIRC= %mir4 and FAQ= %faq
```

Remote events (see later) :

```
2:ON TEXT:xdcc send #1*?::/dcc send $nick c:\temp\serve\mir392.zip | /inc %mir4 1
2:ON TEXT:xdcc send #2*?::/dcc send $nick c:\temp\serve\mircfq23.zip | /inc %faq 1
Alias :

/stats /echo 6 mIRC: %mir4 and FAQ: %faq
```

With these lines I made a small download counter.... not perfect but working ! You can even add some alias like the next one to keep partial statistics : /reset /set %date \$day \$date | /inc %mirctot %mir4 | /inc %faqtot %faq | /set %mir4 0 | /set %faq 0  
/stats /echo 6 Stats mIRC: %mir4 ( %mirctot ) and FAQ: %faq ( %faqtot ) (Set on %date )

When you increase or decrease a non-existent variable it will be created automatically and set to the value you expect it to be. ie :

```
/inc %test1 6 will set %test1 to 6 if it didnt exist before
/dec %test2 7 will set %test2 to -7 if it didnt exist before
```

Note : You have to make sure you always include the % ! This is made to prevent a lot of small possible problems in mIRC's command parsing. This does not limit the ways the variables can be used in at all since you can always use constructions like :

```
1:upme:/inc % $+ $site 1
2:xdcc send #1:/dcc send $nick c:\temp\serve\mir40.zip | /inc % $+ $nick 1
2:xdcc send #2:/dcc send $nick c:\temp\serve\mircfq24.zip | /inc % $+ $nick 1
2:xdcc stats:/notice $nick You downloaded %nick files already since %date
```

[Table of contents](#)

## **8 SOME SAMPLE MIRC SETTINGS (aliases, popups, Remote, etc.)**

Well.... since I got and saw a LOT of questions on all kinds of settings I'll give you a lot of mine here ... and some basic explanation...

On a /whois nick1 the server reports :

Nick1 t.r.vonck@somewhere.nl \* Tjerk Vonck  
Nick1 #mine  
Nick1 irc.somewhere.nl Some University, The Netherlands

#### File/Setup/IRC\_Servers

Most connection problems have their origin in faults in this section.  
Make sure you fill in the correct info in the correct place.

Real Name: Tjerk Vonck  
E-mail: t.r.vonck@somewhere.nl  
(Used internally! USE THE CORRECT ADDRESS!)  
Nick Name: Nick1  
Alternate: Nick2

#### File/Setup/Local\_Info

Local Host: Cheops  
(The name of my PC which I declared myself in my TCP/IP32 winsock)  
IP Address: 130.89.236.62 (The IP number of my PC)  
(I have permanently the same IP number, so I have set 'Always get IP Address a nd  
Local Host on connect' to "OFF". On machines with Dynamic IP Addressing you have  
to set these to ON. )  
IRC Server: (Fill in a nearby public IRC server...)  
Port: 6667 (The standard IRC port number)

#### File/Setup/Options/

I dont need the setup dialog to popup on startup. I have everything set as I normally  
want it already. I like the used servers to be sorted to the top of my list.  
Connections are retried 2 times with a 40 seconds timeout delay.

#### File/Setup/Identd/

An ident server normally is maintained by the Unix machine of your network  
provider... It is a kind of nameserver that guarantees your Identity. Since most  
standalone windows machines are not correctly backed up by an ident service, such a  
server is built into mIRC. More and more IRC servers require you to be identified in  
some way, and they will disconnect you if you're not identified !  
Server is Active (Only if you are normally not identified you need this)  
User ID: t.r.vonck (the part before the @ in your E-mail address)  
System: UNIX (ALWAYS fill in UNIX !! not dos, win or whatever else !!)  
Listen on port: 113 (The standard ident port number)

#### File/Setup/Firewall/

The firewall support in mIRC is new and fairly untested. I'm not behind a firewall so I  
dont use it. If you are behind a firewall your system admin (sysop) can tell you what  
settings you need. If you never even heard about firewalls it is very unlikely you need  
to fill in anything here.

#### Misc/Options/Action Lists/

(Make sure to set these sections to active if you want them to work!)  
Perform: /join #mine | /join #friendly | /flood 12 2 2  
Highlight: mynick,hello,bye

Notify: tinner skala oo7 madonna

## Tools/Aliases

The alias section provides a list of shortcuts. You can make all kinds of aliases... You can shortcut commands as well as things you frequently say. You can put multiple commands in an alias by using the | as a separator.

Instead of typing "/join #mirc" I can do "/j #mIRC" and even "/jm"  
Instead of typing "/kick nick" I use "/k nick"  
Watching the Dutch soccer team I use "/g"  
And I type "/l nick" to do a "/ctcp nick xdcc list" on him.

Check my aliases list to see why and how these aliases work...

```
/f5 /say This text is shown to you on pressing F5
/f6 /dcc send $1 c:\temp\serve\mirc40.zip
/f7 /topic # New topic for the channel you're on.
/nick /say I like $1 better than $me | /nick $1
/home /me says: The mIRC Homepage ??.. "http://huizen.dds.nl/~mirc/"
/lines /say line | /say another line | /say more lines
/op /mode # +ooo $1 $2 $3
/deop /mode # -ooo $1 $2 $3
/j /join $1
/w /whois $?="Who is who"
/k /kick # $1
/hop /part # | /join $1
/jm /join #mIRC
/dbk /mode # -o $1 | /kick # $1 | /mode # +b $1
/l /ctcp $1 xdcc list
/s /ctcp $1 xdcc send $2
/g /say GG000000AAAAAALLLLLLLLLLLLLL!!!!
/i /invite $1 #
/hug /me hugs $1 firmly!
```

## Tools/Popups/Main window (menu is active)

```
Set Away...:/ame is AWAY ( $+ $?="Give a reason" $+ ) | /away Please try later ( $+
$! $+ ) | /timer5 10 900 /me is away ( $+ $! $+ )
Set Back:/ame is back,.. lets start the fun ! | /away | /timer5 off
-
Join
.#mIRC:/join #mIRC
.join...:/join $$?="Give channel name"
Other
.Action...:/me $$?="What are you doing ?"
.Nickname...:/nick $$?="Give your new nick"
.Query...:/query $$?="Who do you want to talk to ?"
.Whois...:/whois $$?="Who is who ?"
.Change Topic...:/topic # $$?="Give new topic"
.List Channels:/list
.Part Channel:/part #
Names
.#mIRC: /names #mIRC
.#Friendly: /names #Friendly
```

```

.names...:/names $$?="Give #channel name"
Invite
.Something...:/invite $$?="Who to invite ?" #
Channel
.Invite Only:/mode # +i
.Not Invite Only:/mode # -i
.Limit...:/mode # +l $$?="Give the max amount"
.No Limit:/mode # -l 0
.Private:/mode # +p
.Not Private:/mode # -p
.Messages:/mode # -n
.No Messages:/mode # +n
.Secret:/mode # +s
.Not Secret:/mode # -s
.Moderated:/mode # +m
.Not Moderated:/mode # -m
Finger
.xhlec@wmin.ac.uk:/finger xhlec@wmin.ac.uk
.Address...:/finger $$?="Give the address"
-
Edit a Note:/run notepad.exe c:\temp\note.txt
News:/run c:\network\winvn\winvn.exe
Netscape...:/run c:\network\netscape\netscape.exe $?="Give URL"
FTP...:/run c:\network\wsftp\ws_ftp.exe $$?="Give site name please"

```

Tools/Popups/Query+Chat window (Menu is active)

```

UWhois:/uwho $1
Whois:/whois $1
Ping:/ctcp $* PING
Version:/ctcp $* VERSION
DCC-Chat:/dcc chat $1
Slap!:/me slaps $1 around a bit with a laaaarge trout

```

Tools/Popups/Channel names list (Menu is active)

```

UWhois:/uwho $1
Private:/query $1
Ping:/ctcp $* PING
Version:/ctcp $* VERSION
-
Control
.Give Ops:/mode # +ooo $$1 $2 $3
.Deop:/mode # -ooo $$1 $2 $3
.Kick:/kick # $1
.Ban:/ban $1
.Deop, Kick, Ban:/mode # -o $$1 | /kick # $$1 | /ban $$1
CTCP
.Ping:/ctcp $1 ping
.Finger:/ctcp $1 finger
.Version:/ctcp $1 version
.Time:/ctcp $1 time
.UserInfo:/ctcp $1 userinfo
.ClientInfo:/ctcp $1 clientinfo
DCC

```

```
.Send:/dcc send $1
.Chat:/dcc chat $1
mIRC CTCP
.mylevel:/ctcp $1 mylevel
.mycmds:/ctcp $1 mycmds
-
Slap!:/me slaps $1 around a bit with a frozen trout
Fish-shop
.Initiate
..Slap trout:/me slaps $1 around a bit with a frozen trout as a mIRC initiation rite ...
..Tickle swordfish:/me tickles $1 a bit with a frozen swordfish as a mIRC initiation
rite ...
.For fun
..Slap trout:/me slaps $1 around a bit with a frozen trout, just for fun ...
..Tickle swordfish:/me tickles $1 a bit with a frozen swordfish, just for fun ...
..Slam dunk:/me slam dunks $1 in a tin of 2 years old tasty surströmming
```

#### Tools/Popups/Double click .....

```
in Status Window does : /notify
in Channel Window does : /channel
in Query Window does : /whois $$1
in Names List does : /uwho $$1
```

#### Tools/Remote/Users

```
10:*!t.r.vonck@wsomewhere.nl
5:*!f.t.a.port@somewhere.nl
4:nick1
4:nick2
4:afriend
4:otherfriend
4:morefriends
3:bot1
3:bot2
=95:*!b.t.gates@pentium.company.com
=95:*!bla@unix1.teched.oc.nl
```

```
Default level: 2
Listening ON
```

#### Tools/Remote/Commands. (Dont bother me with questions about these sections)

(Warning :: Improper settings can crash mIRC with a GPF)

```
1:url:/notice $nick I currently look at $url
1:mama:/notice $nick $read c:\network\mirc\yomama.fun
1:version:/notice $nick This is mIRC. Version 3.64 is available to you on #mirc !!
1:ping:/notice $nick ping ? .. ... .. Humpf ! :>
1:address:/notice $nick Your address is $nick $+ ! $+ $address
1:info:/notice $nick This is the new Winsock irc client mIRC v3.7
1:cmds:/notice $nick Existing cmds are : info, address, msg, opme, mylevel, cmds,
mycmds, part, die, do, xdcc, server, and many more... :> mycmds
1:mycmds:/notice $nick Available to you are : mylevel, cmds, mycmds, info, address.
1:whoami:/notice $nick Please use : cmds, mycmds and mylevel.
```



1:xdcc list:/notice \$nick Sorry no files are offered (or xdcc capabilities exceeded).  
1:a:/notice \$nick This is NOT ws\_irc. \*This\* client wont crash for you.... This is mIRC.  
1:die:/notice \$nick This client won't die for you .... your levels are too low ;-) :>  
mycmds  
1:msg:/notice \$nick Sorry you can't perform msg... your levels are too low :-> :>  
mycmds  
1:opme:/notice \$nick Sorry you can't perform opme... your levels are too low :-> :>  
mycmds  
1:part:/notice \$nick Sorry you can't perform part... your levels are too low :-> :>  
mycmds  
1:server:/notice \$nick Sorry you can't use the server at the moment....  
1:list:/notice \$nick Please use the xdcc commands "/ctcp \$me xdcc [list/send #?]" .  
1:send:/notice \$nick Please use the xdcc commands "/ctcp \$me xdcc [list/send #?]" .  
1:xdcc send:/notice \$nick Sorry DCC limits exceeded .. try later ??  
1:mylevel:/notice \$nick You have command level 1.  
2:xdcc list:/notice \$nick I'm offering the new Winsock irc client mIRC, version 4.0 :>  
lst1  
2:lst1:/notice \$nick #1 mirc40.zip (624K) The 16 and 32 bits versions in one zip :>  
lst2  
2:lst2:/notice \$nick #2 mirc FAQ 24.zip (70K) (Totally mIRC 4.0 ready!) :> lst3  
2:lst3:/notice \$nick Do a "/CTCP \$me xdcc send [#1/#2]" to get a file...(Or /ctcp \$me  
server)  
2:xdcc send #1:/dcc send \$nick c:\temp\serve\mirc40.zip  
2:xdcc send #2:/dcc send \$nick c:\temp\serve\mircfq24.zip  
2:xdcc send #4:/dcc send \$nick c:\temp\serve\mirc-3d.zip  
2:xdcc send:/notice \$nick Sorry, TYPO or you didn't specify a pack number...  
2:server:/fserve \$nick 3 c:c:.txt  
2:mycmds:/notice \$nick Available to you are : mylevel, cmds, mycmds, info, address,  
xdcc, server.  
2:mylevel:/notice \$nick You have command level 2.  
3:mycmds:/notice \$nick Available to you are : mylevel, cmds, mycmds, info, address,  
xdcc, server.  
3:mylevel:/notice \$nick You have command level 3.  
4:mycmds:/notice \$nick Available to you are : mylevel, cmds, mycmds, info, address,  
xdcc, server.  
4:mylevel:/notice \$nick You have command level 4.  
5:opme:/mode \$parm1 +o \$nick  
5:mycmds:/notice \$nick Available to you are : mylevel, cmds, mycmds, info, address,  
xdcc, server, opme.  
5:mylevel:/notice \$nick You have command level 5.  
6:mycmds:/notice \$nick Available to you are : mylevel, cmds, mycmds, info, address,  
xdcc, server, opme.  
6:mylevel:/notice \$nick Your command level is level 6 or higher.  
6:ping:/wavplay c:.wav | /notice \$nick ping ? pang ! ..... DEAD  
10:part:/part \$parms | /notice \$nick I have left channel \$parms \$parms  
10:mycmds:/notice \$nick Available to you are : mylevel, cmds, mycmds, info,  
address, xdcc, server, opme, part, die, do.  
10:mylevel:/notice \$nick You have command level 10.

Tools/Remote/Events. (Dont bother me with questions about these sections)

(Warning :: Improper settings can crash mIRC with a GPF)

1:ON TEXT:mama:#mirc:/msg \$chan \$nick \$read c:\netwerk\mirc\yomama.fun  
1!:ON SERVERMODE:#mirc:-spmilk+tn:/notice \$me Mode change by server

1!:ON OP:#mirc:/notice \$me Watch out \$opnick is opped on #mIRC by \$nick  
1:ON SERVEROP:#mirc:/mode \$chan -o \$opnick | /notice \$opnick Sorry unknown  
SERVEROP's not allowed on #mIRC...  
1:ON TEXT:sjit:#mirc:/kick #mirc \$nick wordkick on 'sjit' | /notice \$nick Hello \$nick,  
you said 'sjit' and that's not allowed on #mIRC..  
1:ON TEXT:=xdcc list:?:/notice \$nick please use "/ctcp \$me xdcc list" But... no files  
are offered at the moment.... | /closemsg \$nick  
1:ON TEXT:=xdcc send:?:/notice \$nick please use "/ctcp \$me xdcc send #?" But... no  
files are offered at the moment.... | /closemsg \$nick  
2:ON TEXT:=xdcc list:?:/notice \$nick please use "/ctcp \$me xdcc list" | /closemsg  
\$nick  
2:ON TEXT:=xdcc send #1:?:/dcc send \$nick c:\temp\serve\mirc40.zip | /notice \$nick  
Please next time use '/CTCP bla bla bla..' ?? | /closemsg \$nick  
2:ON TEXT:=xdcc send #2:?:/dcc send \$nick c:\temp\serve\mircfq24.zip | /notice  
\$nick Please next time use '/CTCP bla bla bla..' ?? | /closemsg \$nick  
3!:ON JOIN:#mirc:/auser 3 \$nick  
3:ON OP:#mirc:=  
3:ON MODE:#mirc:=  
3:ON SERVEROP:#mirc:=  
3!:ON DEOP:#mirc:/mode \$chan +o \$opnick | /notice \$nick Hey! \$opnick is my  
friend!  
+3:ON PART:#mirc:/ruser \$nick  
+3:ON KICK:#mirc:/ruser \$knick  
+3:ON QUIT:/ruser \$nick  
+3:ON NICK:/ruser \$nick | /auser 3 \$newnick  
4!:ON KICK:#mirc:/kick \$chan \$nick \$knick is my friend! | /invite \$chan \$knick |  
/notice \$nick \$knick is my friend!  
4!:ON DEOP:#:/mode \$chan -o \$nick | /mode \$chan +o \$opnick | /notice \$nick Hey!  
\$opnick is my friend!  
4:ON NICK:/auser 4 \$newnick  
5!:ON JOIN:#mirc:/pop 9 \$chan \$nick | /auser 4 \$nick  
95:ON JOIN:#mirc:/mode \$chan +b \$nick | /ban \$chan \$nick 3 | /kick \$chan \$nick  
sjitlisted | /notice \$nick I'm sorry but you're not welcome to \$chan

[Table of contents](#)



