MCF4AmIRC

COLLABORATORS

	TITLE : MCF4AmIRC		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		March 1, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

MCF4AmIRC

1.1   MCF4AmIRC Guide, An Integrated ARexx System for AmIRC.

                                     MCF4AmIRC V_4.13 (26.12.96)
                    Copyright 1996 Donald T. Becker

              Disclaimer

              Description/What's New

              Requirements

              Pre-Install Testing

              Installation

              Usage

              TODO

              History/Bug Fixes

              About the Author...

1.2   DISCLAIMER:

DISCLAIMER:

        The program contained in this archive is copyrighted by
Donald T. Becker and is available as Freeware Software.
It may be distributed only under the following conditions:
- No charge is made to any of the files in this archive.
- Programs and Documentation are not modified in any way.
- The Package is always distributed in its complete form consisting of
84 files:
  10 icons: "CleanUp_MCF.rexx.info", "DalHelper.Doc.info", "ENV.info",
    "Install_MCF.info", "MCF.info","MCF4AmIRC.ReadMe.info",

      "MCF4AmIRC.guide.info", "MCF4AmIRC.info", "Rexx.info",
      and "SetUp_MCF.rexx.info".
   5 files in dir MCF4AmIRC: "DalHelper.Doc", "Install_MCF",
      "MCF4AmIRC.guide", "MCF4AmIRC.readme", and "SetUp_MCF.rexx".
   1 file in dir MCF4AmIRC/ENV: "MCF_JOIN.AMIRX".
   37 files in dir MCF4AmIRC/Rexx: "CTCP_Sound.AMIRX", "DalHelper.AMIRX",
      "FAQ.AMIRX", "JSound.AMIRX", "MCF.AMIRX", "MCF_Alias.AMIRX",
      "MCF_AU.AMIRX", "MCF_Auto.AMIRX", "MCF_ClientInfo.AMIRX",
      "MCF_DO.AMIRX", "MCF_DU.AMIRX", "MCF_Encode.AMIRX", "MCF_Finger.AMIRX",
      "MCF_FLIST.AMIRX", "MCF_FSEND.AMIRX", "MCF_FServ.rexx", "MCF_GETSOUND.AMIRX",
      "MCF_HELP.AMIRX", "MCF_K.AMIRX", "MCF_KB.AMIRX", "MCF_MDO.AMIRX",
      "MCF_MKB.AMIRX", "MCF_OK.AMIRX", "MCF_OKB.AMIRX", "MCF_OP.AMIRX",
      "MCF_PING.AMIRX", "MCF_PROTECT.AMIRX", "MCF_SAbuser.AMIRX",
      "MCF_Silence.AMIRX", "MCF_SOUND.AMIRX", "MCF_STATUS.AMIRX",
      "MCF_UI.AMIRX", "MCF_USERS.AMIRX", "MCF_VERSION.AMIRX",
      "MFAQ.AMIRX", "NFK.AMIRX", and "Say.AMIRX".
   31 files in MCF4AmIRC/MCF: "MCF.Cmds", "MCF.Encrypt", "MCF.FServ",
      "MCF_GChans", "MCF.GNicks", "MCF.Greets", "MCF.Ops", "MCF_Pings",
      "MCF.PNicks", "MCF.Protect", "MCF.StartUp", "MCF_AU.help",
      "MCF_AUTO.help", "MCF_DO.help", "MCF_DU.help", "MCF_FLIST.help",
      "MCF_FSEND.help", "MCF_HELP.help", "MCF_K.help", "MCF_KB.help",
      "MCF_MDO.help", "MCF_MKB.help", "MCF_OK.help", MCF_OKB.help",
      "MCF_OP.help", "MCF_OPTS.help", "MCF_QUIT.help", "MCF_RC.help",
      "MCF_RU.help", "MCF_STATUS.help", and "MCF_USERS.help".

- Every form of public distribution is allowed and encouraged, but no fee
  can be charged for this program except for, possibly, the cost of
  magnetic media and/or disk duplication and shipping.
- Inclusion in PD software libraries such as Fish Disks and AmiNet CD's
  is allowed (and encouraged), provided the fees charged for these disks
  are comparable with current charges for Freeware Software.
- Inclusion of this software in a future release of AmIRC is allowed.
- Inclusion of this software in a Cover Disk Magazine is allowed
  after receiving authorization from the author.
- This program cannot be distributed in any commercial product without
  the written consent of the author.

By copying, distributing, and/or using these programs, you indicate your
acceptance of the above rules.

        THE PROGRAM IS SUPPLIED 'AS IS' : THE ENTIRE RISK AS TO THE
QUALITY OF THE PROGRAM IS TO THE USER. IN NO EVENT WILL THE AUTHOR BE
LIABLE FOR DIRECT OR INDIRECT DAMAGE OR LOSS RESULTING FROM THE USE/MISUSE
OF THIS PROGRAM.

P.S.
        The only thing I will guarantee is that this program performed
to expectation whenever I made use of it. If it doesn't work for you,

            IT ISN'T MY FAULT....

But if it doesn't work for you or you run into any problems,
I would appreciate being notified of the situation so that I may

correct it in a future update.

1.3 DESCRIPTION:

DESCRIPTION

    MCF is a set of Integrated ARexx scripts which will add many features
    to your AmIRC sessions. First and foremost is Auto-Ops. Also included
    are Remote User Commands which can be executed by authorized users
    listed in your Ops file. Among these are commands like OP (remote
    user OPping someone), DO (remote user De-OPping someone), K (remote
    user Kicking someone from the channel), KB (remote user KickBanning
    someone from the channel), MDO (performs a Mass De-OP of all channel
    OPs except for the remote user and the MCF user), and MKB (performs
    a Mass Kick Ban of multiple users with the same UserHost. ie: Clones).
    Also available are STATUS and HELP. An extra feature is a File Server
    commands. These are FLIST (remote user request DCC of current file
    list) and FSEND (remote user requests DCC of a file from the file list.
    MCF control functions include AUTO (turns the Auto-OPs, FileServer, and
    Remote Commands stitches on and off, and change the ChannelOp character),
    AU (add/update a user entry to the Ops file), DU (remove a user entry
    from the Ops file), RC (reload command function list), RU (reload Ops
    list), and USERS (displays entries in Ops file). External to MCF is
    "MCF_FServ.rexx". This script is used to generate the files used with
    the file server functions.

    Version 4.13 should no longer open a play requestor from
    Play16. If the requested sound does not exist as a FILE,
    MCF will not attempt to play it.

    Version 4.12 has MCF Configuration File(s) validation.
    If a file does not exist, MCF will tell you which one
    is missing and continue to execute instead of Crapping out.

    Version 4.11 has 1 new function: MCF_Auto.AMIRX.
    This script is a GUI Front-end for setting %AUTO variables.
    Changed the GUI procedure used by CTCP_Sound.AMIRX when
    the user has multiple directories assigned as his SOUNDDIR.
    Fixed DCCGET bug that occurred with version 4.10.

    Version 4.10 has 1 new function: MCF_SAbuser.AMIRX.
    This script will exclude a user from being able to make
    your computer speak.
    Added new %AUTO variable: PSOUND. When = Y MCF will play
    sounds you have that you send to other users.

    Version 4.9 has 2 new CTCP functions: ClientInfo and Finger.
    MCF_Finger.AMIRX requires AmiTCP:bin/Finger for it to respond
    to a CTCP Finger request.
    Fixed a problem in the MCF_HELP script.
    DalNet updated versions so I had to change how the scan in the
    flood protection for DalNet Services was done.
    Added code to MCF to help guarantee that redirected dos commands
    will NOT execute on your machine.

Added Speech and Speech on/off auto switch.

Version 4.8 no longer opens the MSGPort "SFPROTECT.?".
Status and Help commands were altered to handle this without
flooding yourself off of the server. This is accomplished by
only processing 1 at a time and ignoring all other Status and
Help requests while 1 is processing.
MCF Flood Protection routine should no longer ignore
Service@Dal.net anymore.
MCF will DeCode an Encoded Topic upon Join a channel that
has an Encoded Topic.

Version 4.7 has all configuration files moved from the
ENV: & ENVARC: dirs to the MCF: dir.
A bug that crept into the join script when greets was
added has been fixed.
2 new options have been added to the GREETS %AUTO variable.
P will send the Auto-Greets to the user by way of PRIVMSG.
C will only Auto-Greet the user is the channel being joined
   is listed in the file: MCF:MCF.GChans.
CTCP_Sound.AMIRX is now more Win95 compatible.
MCF Aliases are now installable thru the install script.
Install script now moves configurations files from ENV: to
MCF: and deletes 1 obsolete script from ENV:.
Added new CTCP function: MCF_PING.AMIRX.
Random Ping responses are stored in MCF:MCF.Pings,
Nick Specific Ping Responses are in MCF:MCF.PNicks.
Valid Ping Variables are %n = Your Nick, %u = User's Nick.
The following is now valid for Greets and Pings:
/S = send Sound. If you do: /S soundfile.wav
you can also add an Action (/ME) or a msg after the sound.

Version 4.6 has a bug fixed in the !Nick Auto-Request
function. Flood Protection has been Re-Written (again).
Greets have been added to the join function.
Random Greet entries are placed in "MCF:MCF.Greets".
Nick specific entries are placed in "MCF:MCF.GNicks".
Variables allowed are: %c = Channel, %n = Your Nick,
and %u = User's Nick. To greet with an action, use
/me at the beginning of the greet message.

Version 4.5 has a new function: "DalHelper.AMIRX",
and a new feature: "NFK.AMIRX" (used by "MCF_MKB.AMIRX").
A new feature has been added to MCF_GETSOUND.AMIRX.
Just enter: !Nick List in a channel and that Nick will
generate a list file of his sounds and send you the file.
Added Topic option to MCF_Encode.AMIRX.
Added several more functions to DalHelper.AMIRX.
Added DalHelper Configuration to Install_MCF.
Fixed Decryption bug in MCF.AMIRX.
Fixed bug in the %AUTO USE/SAVE routines.
Change auto-request feature of MCF_SOUND.AMIRX
to only fetch a file once, regardless of the number
of times requested.
Added a GUI for sending sounds using "CTCP_Sound.AMIRX".
Added Auto-Get feature for Auto-Requested sounds.

Version 4.4 has Extra kick power added to MKB.AMIRX.
Added new function for users of Dal's NickServ & ChanServ:
 DalHelper.AMIRX

Version 4.3 is no different from 4.2 except that the
scripts have "Rexx:" hard-coded internally.

Version 4.2 has 2 new functions: %OK "MCF_OK.AMIRX",
and %OKB "MCF_OKB.AMIRX".

Version 4.1 has had the main script "MCF.AMIRX"
re-structured to (hopefully) execute quicker.
The hard-coded encryption schemes were removed from
both "MCF.AMIRX" and "MCF_Encode.AMIRX". It now
resides in a file in the MCF: dir named "MCF.Encrypt".
Added new AUTO variable "AJOIN". This variable is
used to decide if you Auto-Join a Channel on an
INVITE or not.
The "MCF_VERSION.AMIRX" now has the URL of the
MCF Home Page.

Version 4.0 is nearly a complete re-write of the main and
all remote command scripts. Almost all remote commands
are now run as background tasks so that the main script
(MCF.AMIRX) can continue it's scanning a lot quicker.
Added new %AUTO variable called "NOSOUNDS". This has 4
different modes which will control your response to a
"SoundFile not Found" condition from a CTCP SOUND request.
Added 2 new CTCP functions. "MCF_VERSION.AMIRX" responds
when someone does a CTCP VERSION on you.

"MCF_GETSOUND.AMIRX" is a function that is compatible
with another script out there called "CTCP.AMIRX". It
will DCC SEND the soundfile to the user who sent you
the CTCP GETSOUND request.
Added support for the auto-fetching of soundfiles
in the following format: !Nick file.wav.

Added a size scanning routine to the Flood Protection
procedure. This helps prevent floods from CTCP ECHOs
and some others quicker than it would by just scanning
for flood hits. Added a new script and 2 icons that
can be used to test MCF before you install it.
Added the use of the server Silence command along
with the AmIRC Ignore command.

Added code to handle all remote commands uniquely.
New Format: %Nick CMD Options
 ie. %StarDustr OP MyNick
The original format still works as well: %CMD Options.

Added new Extra command: JSound.AMIRX
This function will send a CTCP SOUND request to
a channel before(default) or after you join it.

Added new Flood protection command: MCF_Silence.AMIRX
This function is to be manually used to ignore

a user if you suspect him of getting ready to
flood you.

Added support for Cybeonix (version 1.3.3) message
encryption decoding.
Added new Extra command: MCF_Encode.AMIRX
This function will encrypt messages and send them
to the channel, a user, or a specified channel.
This function also supports 10 other encryption
methods exclusive to MCF.

Added new NOSOUNDS options to allow similar settings
when using AmIRC 1.5Beta, but will not attempt to
play the sound (since AmIRC 1.5Beta now will play it)
To get the same responses as before, just add 5 to
your current NOSOUNDS setting.

Version 3.1 has 2 new commands. 1 is an Extra command
called "Say.AMIRX". It allows you to send Highlighted,
Reversed, and/or Bold text to the server. 2 is a new
level 99 command called "%OPTS". This will display to
you the current changeable environment settings in use.

Version 3.0 now supports ENV variable configuration files.
This will make future upgrades easier to install.
Also included is more Flood Protection. Now handles TEXT and PRIV.

Version 2.2 has a new script to make it easier to send level 99
commands to MCF.

Version 2.1 has 9 more help files and the Status function has been
rewritten..

As of Version 2.0, MCF now has FLOOD Protection and CTCP Sound Support,
as well as a few extras.

1.4 REQUIREMENTS:

REQUIREMENTS

An Amiga with AmIRC installed and running correctly.
(The ARexx interface must be turned on thru MUI)
ARexx installed and running correctly.
(a path to rexxc and an assign to Rexx: are required)
reqtools.library and rexxreqtools.library are
    required to use the GUI portion of "CTCP_Sound.AMIRX".
If AmIRC works, and ARexx if up and running,
    these scripts should then run fine.

1.5 Pre-Install Testing

2 new Icons are in the MCF4AmIRC directory that can be
used to allow you to test MCF without installing it first.
Click on the "SetUp_MCF.rexx" icon and this will change
REXX:, ENV: and MCF: to point to the de-arced archive
directories. You could then start AmIRC and test MCF to
see if it is working correctly for you. When done testing,
click on the icon "CleanUp_MCF.rexx" and this will reset
your assignments.

btw - SetUp_MCF.rexx creates the "CleanUp_MCF.rexx" script,
      So don't go looking for it before running SetUp_MCF.rexx.

1.6 Installation

INSTALLATION

  Installation of MCF is fairly simple. Just click on the
- Install_MCF icon and go thru the standard Installer Procedure.
- Or, if you like to do things manually, Copy all files in the
- MCF4AmIRC/Rexx directory to your Rexx: directory. Copy all files
- in the MCF4AmIRC/ENV directory to ENV: and ENVARC:. Either make
- or select a directory and copy all the files from MCF4AmIRC/MCF
- into it.
- Add the following assign to your user startup:

  Assign MCF: WhereEverYouPutThem

    Run AmIRC and goto Settings->Setup->Events

    Enter ENV:MCF_JOIN %p in the Rexx Command field for the JOIN Event.
         Note   %p is a lowercase p.
    * Update Notice *   Delete REXX:MCF_JOIN.AMIRX, it is now in ENV:.

    MCF should come up running when you join your first channel.

    MCF can be manually started by typing the following before you
    join your first channel:

 /Rx MCF NOOPS/S FSERV/S NOSOUNDS/S NORCMDS/S

    Where the NOOPS switch turns off the Auto-Ops and OP functions, the
    FSERV switch turns on the File Server features, the NOSOUNDS switch
    turns of the CTCP Sound function, and the NORCMDS switch disables
    all Remote Commands. (Level 99 commands not included)

    Or you can edit the MCF_JOIN.AMIRX file and insert any combination
    of the switches between MCF and the "   to the following line:

            "say /Rx MCF"

    The above switches may now be set thru the file MCF.Startup which is
    located in your ENV: directory. Just set the value on each line to
    the value you wish for a default setting. Use upper case Y and N

for all flag settings. Sound directory and player may use mixed case.
** Using switch(s) when starting MCF will override those ENV setting(s).

*** Installation WARNING ***
   Due to some defect in the Amiga OS, things do not always
   work right when there are multiple assignments to the
   logical device REXX:.   Therefore, to make sure this
   and other AREXX scripts work correctly, there should be
   only 1 logical assignment to REXX:
   (I personally made a directery called Rexx and use this
      for all my AREXX scripts.)

   Also, due to the differences in different user's system
   configurations, MCF does not always work right. If once
   MCF loads and you do a CTCP VERSION on yourself and don't
   get a PRIVMSG specifying the version of MCF you have,
   please email me and we can go thru a couple of procedures
   to try and get it running for you.

*** In-compatibilites ***

   It seems that MCF and CTCP.AMIRX tend to conflict in a way
   that prevents MCF from running correctly.

   For some strange, unkown reason, MCF runs but doesn't work
   for some people when started from AmIRC. If it loads but
   doesn't seem to be doing anything, try to start it from a
   shell and attach it to the AmIRC client by adding the PORT
   name as command-line input. The PORT name is in the upper
   left corner of your AmIRC window. It will be   AMIRC.#
   where # is a number. Start MCF from a shell for the first
   AmIRC client like so: Rx MCF.AMIRX AMIRC.1

1.7 Usage

                Before any commands will work, MCF must be running. How this ←
              is
   accomplished is described in
              Installation
                  The commands are broken down into 5 basic levels.

   Remote User Commands:
              Levels 00 thru 20
                 Remote User Commands:
              Levels 30 thru 60
                 MCF System Commands:
              Level 99 only
                 MCF CTCP Commands:
              CTCP Events that are Handled
                 MCF FLOOD Related Commands:
              Internal/External Commands
                 Miscelanious MCF Commands:
              Extra Commands

1.8   Commands for Levels 00 thru 20

All commands of this class may be executed by passing the command to
MCF by using: /msg MCFUserNick %Command Args.
Or by a remote user who types the command in the channel display window
in either of the following formats:
%Command Args   or   %MCFUserNick Command Args.

%STATUS   Lists a User's Authorization level and Available Remote Commands
          Usage is: %STATUS
          MinUser Level: 00

%HELP   Lists any available help information.
          Usage is: %HELP or %HELP Funct   (Where Funct is a Remote
                                                User Function)
          MinUser Level: 00

The other commands of this class are File Server commands. For these to
work, you must first have edited the file "MCF:MCF.FServ" and entered
all the Directorys to use for building the file list. After that is done,
execute the following Rexx command: Rx MCF_FServ
This will build the files required by the File Server functions.

** Known BUG ** For some reason, typing /DCC SEND with a filename that
          contains 1 or more spaces does not seem to work. Therefore,
          %FSEND requests for files with spaces in thier name won't work.

%FLIST   Requests a list of files available for remote DCC downloading.
          Usage is: %FLIST
          MinUser Level: 10

%FSEND   Request a file be sent to a remote user via DCC downloading.
          Usage is: %FSEND #?.   (where ? is the number of the file from
          the list you want.   The # preceeding the filenumber and a period
          after are required.
          MinUser Level: 20

Note   You can give unrestricted access to the FSERV functions by editing
          the "MCF.Cmds" file and replacing the 10 and 20 with 00. After
          this, either Re-Load Commands (%RC) or Quit (%QUIT) and restart MCF.

1.9   Commands for Levels 30 thru 60

All commands of this class are channel oriented commands and can only be
executed by a remote user typing the command in the channel display window
in either of the following formats:
%Command Args   or   %MCFUserNick Command Args.
None of these commands can be used against the user running MCF.

%K   Kicks a user from the channel..
          Usage is: %K UserNickName reason (reason is optional)
          MinUser Level: 30

%OP   Give OPs to a user.

```
              Usage is:        %OP UserNickName
                            (UserNickName is optional. Requestor will be OPped)
              MinUser Level:   40


  %DO       Remove OPs from a user.
            Usage is:        %DO UserNickName
            MinUser Level:   40


  %KB       Kick and Ban a user from the channel.
            Usage is:        %KB UserNickName Reason (Reason is optional)
            MinUser Level:   40


All users defined with a level of 50 or greater will be Auto-OPped if enabled.


  %MDO      Remove OPs from all users on the channel, excluding
            the user running MCF and the user requesting %MDO.
            Usage is:        %MDO
            MinUser Level:   50


  %OK       OP a user, then Kick the user from the channel.
            Usage is:        %OK
            MinUser Level:   50


  %MKB      Kick and Ban all users from the channel who are on the same UserHost.
            Usage is:        %MKB Arg Reason (Reason is optional)
                            (Arg can be a UserNickName or UserHost of a user)
            MinUser Level:   60


  %OKB      OP a user, then Kick & Ban the user from the channel.
            Usage is:        %OKB
            MinUser Level:   60
```

1.10   MCF System Commands

```
All commands of this level must issued via a /MSG to the user running MCF.
 These commands are not to be executed by a remote user.  Only the user
  of MCF should have a level 99 authorization.


%AUTO  Use to change control switch settings.
     Switch: ?              This will bring up the GUI Front-end for setting  ←
        values.
     Switch: SPEAKPLAYER Value:  Name of Speech Player (including drive and  ←
        dir path).
     Switch: SOUNDPLAYER Value:  Name of Sound Player (including drive and  ←
        dir path).
     Switch: SOUNDDIR    Value:  Directory/Assignment where Sounds are stored ←
        .
     Switch: DCCDIR      Value:  Directory/Assignment where Auto-Get will  ←
        Download to.
     Switch: AJOIN       Values: Y/N   Function: Enable/Disable Auto-Join  ←
        function.
     Switch: ASEND       Values: Y/N   Function: Enable/Disable Auto-Sending  ←
        of requested Sounds.
```

```
            Switch: DCCGET      Values: Y/N   Function: Enable/Disable Auto-Get  ←
               function.
            Switch: FSERV       Values: Y/N   Function: Enable/Disable Fileserver  ←
               functions.
            Switch: GREETS      Values: Y/N   Function: Enable/Disable Auto-Greets  ←
               function.
            Switch: OPS         Values: Y/N   Function: Enable/Disable Auto-OPs  ←
               function.
            Switch: RCMDS       Values: Y/N   Function: Enable/Disable Remote User  ←
               commands.
            Switch: SOUNDS      Values: Y/N   Function: Enable/Disable CTCP Sound  ←
               function.
            Switch: PSOUND      Values: Y/N   Function: Enable/Disable Playing of  ←
               sent sounds.
            Switch: SPEAK       Values: Y/N   Function: Enable/Disable Speech  ←
               function.
            Switch: NOSOUNDS    Values: 0-9   Function: Changes response to "  ←
               SoundFile not Found".
                  0 = No Response, 1 = Notify Requestor,
                  2 = Notify You,  3 = Issue CTCP GETSOUND to requestor.
                  4 = Issue !Nick sound.wav to the requestor.
            The following settings should be used with AmIRC 1.4 Beta.
                  5 = No Response, 6 = Notify Requestor,
                  7 = Notify You,  8 = Issue CTCP GETSOUND to requestor.
                  9 = Issue !Nick sound.wav to the requestor.
            Usage is:      /MSG MCFUserNick %AUTO Switch Value

  %OPTS  Use to Display the current settings of MCF.
            Usage is:      /MSG MCFUserNick %OPTS

  %RC    Use to Re-Load the current Commands List. (For use after editing "MCF:  ←
     MCF.Cmds")
            Usage is:      /MSG MCFUserNick %RC

  %RU    Use to Re-Load the current list of Users. (For use after editing "MCF:  ←
     MCF.Ops")
            Usage is:      /MSG MCFUserNick %RU

  %USERS Use to List all Users and thier Status Level currently listed in "MCF:  ←
     MCF.Ops".
            Usage is:      /MSG MCFUserNick %USERS

  %QUIT  Use to Terminate MCF.
            Usage is:      /MSG MCFUserNick %QUIT

    Last of the MCF System Commands are the commands used to Add/Update/Delete  ←
  entries
    in your MCF:MCF.Ops file. All entries in the file consist of the User's  ←
       default
    Status Level (a 2 digit number, 00-99), the User's Nick, UserName, HostName
    and any Priv Channels and their authorization level in the following
    format:  20 Nick!UserName@ Static.Portion.of.UserHostName #chan1 60 #chan2 50
    Nick must be followed by a !  and UserName must be followed by a @
    If the User logs in using a Dynamic connection, a portion of the UserHostName  ←
       always
    remains the same.  This is the portion to enter for a UserHostName.
```

```
%AU    Use to Add/Update User entries in the MCF:MCF.Ops file.
       Usage is:
        /MSG MCFUserNick %AU Nick!User@ Static.Portion.of.UserHostName ?? #Chan ↩
           ??
           (?? being the Status Level you wish to assign to this user,
            and any Privledged channels and their authorization after it.)
       Or, to add a PRIV channel to the OPs file, Usage is:
        /MSG MCFUserNick %AU PRIVCH #channelname

%DU    Use to Delete User entries from the MCF:MCF.Ops file.
       Usage is:      /MSG MCFUserNick %DU ???  (??? being the minimum amount of
                                an entry in the MCF:MCF.Ops file which is unique)
```

1.11 MCF CTCP Commands

```
Following are a list of CTCP Events that are currently handled by
 MCF.  This means the event will be responded to in 1 way or another.

CLIENTINFO This event is handled by MCF_ClientInfo.AMIRX.  It will
           return to the requestor the CTCP Functions currently
           supported by MCF.

FINGER     This event is handled by MCF_Finger.AMIRX.  It requires
           AmiTCP:bin/Finger for it to function.  It will then
           return to the requestor the standard finger information
           about user logged on.

GETSOUND   This event is handled by MCF_GETSOUND.AMIRX.  It will
           result in you  DCC Sending the requested sound file
           to the user issuing the GETSOUND function.

PING       This event is handled by MCF_PING.AMIRX.  It will
           send a response to the user who pings you.  the
           response is determined by data in files:
           MCF:MCF.Pings   and   MCF:MCF.PNicks.

SOUND      This event is handled by MCF_SOUND.AMIRX.  If configured
           right, it will play the requested sound file on your
           machine.  If you don't have it, your response will be
           deterimed by the current setting of AUTO variable "NOSOUNDS".

VERSION    This event is handled by MCF_VERSION.AMIRX.  It will
           send a PRIVMSG to the requestor telling them what
           Version of MCF you are currently running.

!Nick      This is not a not a true CTCP command or function,
           but if received either in the channel or by way of
           PRIVMSG to you, it will translate itself into a
           CTCP GETSOUND request to perform the same function.
           !Nick SoundFile.wav  will request the file "SoundFile.wav"
           from the user.
           !Nick List  will create and send a list of sounds
           that  Nick  has in his SOUNDDIR.
```

1.12   MCF FLOOD Related Commands

4 of the 6 Flood Protect functions are internally executed by MCF.AMIRX and
are not to be manually executed by the user. The Silence and UnIgnore
commands are manually executed by you when you wish to remove an ignore
entry that was made by the Auto-Ignore feature of MCF_PROTECT.AMIRX, or
add/remove a user to manual silence list.

MCF_PROTECT.AMIRX: This is the main Flood Protection script. This script
            is executed when MCF is started, it is not for manual execution.
            It receives its info from the MCF_QProtect script by way of a
            MessagePort. This script currently scans for "PRIV", "CTCP",
            and "TEXT" type of Floods. When MCF_PROTECT.AMIRX count of hits
            exceeds the specified MAX allowed for that type of message
            within the MAX allowed time, an /IGNORE if issued for all
            users at that Domain (unless Domain is listed as an OMIT entry).
            To change the Max hits/times and add/delete Omission Domains,
            edit ENVARC:MCF.Protect and change the desired maxes to the
            value you wish to have for that type of flood.

MCF_Silence.AMIRX: This command is to be manually executed by you
            when you suspect a user is about to flood you.
            Usage is: /Rx MCF_Silence Nick   to silence a user
                 or: /rx MCF_Silence Nick - to un-silence a user.
            For ease of use, make the following Alias:
            SI /Rx MCF_Silence %p   Passed parameter is Nick.
            US /Rx MCF_Silence %p -   Passed parameter is Nick.
            You may also want to make 2 buttons to perform the same
            functions under Settings -> Setup -> Actions.

MCF_UI.AMIRX: This is a command that is manually executed by you
            when you wish to UnIgnore a previously Auto-Ignored User.
            Usage is: /Rx MCF_UI Nick/Domain
            For ease of use, make the following Alias:
            UI /Rx MCF_UI %p. Passed parameter can be either the Domain or
                   a Nick of that Domain.

NFK.AMIRX: This script is used by "MCF_MKB.AMIRX" during a
            Mass Kick Ban of Clones. It will catch any Clones that change
            their Nick during the MKB process.

1.13   Extra Commands

All these commands are manually executed by you except for speech.
 Add Aliases to AmIRC to make use of these commands easier.

Speech: Speech is handled by the main MCF script (MCF.AMIRX).
            To turn Speech on, you must have a SpeechPlayer (SPEAKPLAYER)
            assigned and the Speech switch (SPEAK) set to Y. Speech will
            execute when MCF sees a @ in column 1 of the display. If
            SAY (default) is used for the SpeechPlayer, it will handle the
            standard speech parameters for changing pitch and other variables.

If MCF sees   @Nick   and the nick is a nick belonging to
another user on that channel, MCF will NOT speak that line to you.
MCF will Speak a line that is MSGed to you, @Nick to you, and
@any-text.

MCF_Auto.AMIRX: This script is used by MCF as a GUI Front-end for setting
values of all MCF Startup variables. To use, just pass a
question mark (?) to the %AUTO function instead of a switch
name and it's new value.

MCF_SAbuser.AMIRX: This script is used to add/delete users from your
Speech Ignore List.
Usage is: /Rx MCF_SAbuser Nick or /Rx MCF_SAbuser - Nick
                 (Using - before nick deletes the user from the list)
For ease of use, make the following Alias:
SA /Rx MCF_SAbuser %p. Passed parameters are Nick or - and Nick
   Also, you may add to Events in Setup, the following button:
   Sound   /Rx MCF_SAbuser %u

CTCP_Sound.AMIRX: This script is used to send CTCP SOUND to other users
To Configure: Edit MCF.AMIRX and change the values following
the variables "SoundDir" (where your sounds are located) and
"SoundPlayer" (your soundplayer). Current defaults are:
SoundDir = "Sounds:"   SoundPlayer = "DH0:C/Play16".
Usage is: /Rx CTCP_Sound SoundFileName Destination
GUI Usage: /Rx CTCP_Sound ? Destination
                  (Channel is default if Destination not entered)
For ease of use, make the following Alias:
S /Rx CTCP_Sound %p. Passed parameters are SoundFileName and ←
   Destination.
    Also, you may add to Events in Setup, the following button:
    Sound   /Rx CTCP_Sound ? %u

DalHelper.AMIRX: This script is for use by any user who is a channel
founder, an SOP, an AOP, or has a Registered Nick on Dalnet
using Dal Services: "ChanServ" or "NickServ"
Usage is: /Rx DalHelper C/N Command
Got ease of use, make the following Aliases:
CS /Rx DalHelper C %p   Sends commands to ChanServ.
NS /Rx DalHelper N %p   Sends commands to NickServ.
Valid commands are as follows:
ChanServ: IDentify, HElp, INfo, MDeop, MKick, and UNban.
NickServ: IDentify, GHost, HElp, INfo, RECover, and RELease.
All commands may be entered as just the first 2 letters instead
of the whole word (except for RELease and RECover).. For more
help on each command, connect to Dalnet and then do the following:
/CS HElp [Command] and /NS HElp [Command].

FAQ.AMIRX: This script is used to send a file to a channel.
Usage is: /Rx FAQ FaqDir FaqFilename
For ease of use, make the following Alias:
FAQ /Rx FAQ FaqDir %p FaqDir is the director where your FAQs reside ←
     ,
and FaqFileName is the name of the FAQ to be printed to the channel.

MCF_Alias.AMIRX: This script is for use in sending yourself level 99 ←
    function

```
                commands.  It will send the data to whatever your current Nick is.
                Usage is:   /Rx MCF_Alias %99-level-function arguments for requested ←
                    function
                For ease of use, make the following Aliases:
                MQ     /Rx MCF_Alias %%QUIT      Quits MCF.
                AUTO   /Rx MCF_Alias %%AUTO %p  Execute %AUTO function for setting ←
                    flags.
                AU     /Rx MCF_Alias %%AU %p     Add User to Users list.
                DU     /Rx MCF_Alias %%DU %p     Remove User from Users list.
                OPTS   /Rx MCF_Alias %%OPTS      Lists current running configuration.
                RU     /Rx MCF_Alias %%RU        Reload Users. (Use after manual OPs ←
                    updates)
                MU     /Rx MCF_Alias %%USERS     Lists Users in MCF:MCF.Ops file.

    MCF_Encode.AMIRX:    This script is used to send a encrypted messages to
                the channel, users, or a selected channel.
                Usage is:   /Rx MCF_Encode [-mode] [/nick|channel|Topic] messagetext
      Mode is a number from 0 to 9.  If entered, 1 of 10
                    new encryption methods will be used.  If not entered,
                    Cybeonix 1.3.3 encryption is performed.
                    If a nick or channel is entered, the encrypted
                    message will be sent to that user or channel.
                    if Topic is entered, the Topic will be Encrypted.
                    If not entered, the encrypted message will be
                    sent to the current channel.
                For ease of use, make the following Alias:
                X       /Rx MCF_Encode %p

    MFAQ.AMIRX:          This script is used to send a file to a User by way of / ←
       MSG
                Usage is:   /Rx FAQ FaqDir FaqFilename UserNick
                For ease of use, make the following Alias:
                MFAQ /Rx MFAQ FaqDir %p  FaqDir is the director where your FAQs ←
                    reside,
                FaqFileName is the name of the FAQ to be sent, and UserNick is the ←
                    Nick
                of the user that you want to /MSG the FAQ to.

    JSound.AMIRX:        This script is used to join a channel and
                then send a CTCP SOUND to the channel.
                Usage is: /Rx JSound #Channel Sound.wav (sound.wav is optional)
                For ease of use, make the following Alias:
                JS /Rx JSound %p    If sound.wav is not entered, the default
                sound.wav coded in JSound.AMIRX will be played.  Edit and
                change the sound.wav at  EntrySound  if you desire.

    Say.AMIRX:           This script is used to add special attributes to
                your text.  Attributes like:  Bold, Underscore, and Reverse.
                Usage is:  /Rx Say message text with special attributes.
                To start BOLD         use        \b
                To start REVERSE      use        \r
                To start UNDERSCORE   use        \u
                To end, use the same character you used to start the
                special attribute.
                For ease of use, make the following Alias:
                Say /Rx Say %p
```

1.14 TO DO...

Things to Do in the Future!

```
 M C F 4 A m I R C

     Wait for user response and see what would be best to add next.
     Re-write the Flood Protection routine (MCF_Protect.AMIRX)
       in "C" or Assembly to increase protection speed.
     Add an interactive fileserver that is accessed thru a
       CTCP XDCC command.
     Re-structure MCF Install procedures. Modularize and place
       subroutine install procedures in an Install dir.
```

1.15 History and Bug Fixes....

List of Bug Fixes and Enhancements.

```
26.12.1996 - Fixed a bug in MCF_SOUND.AMIRX.  MCF should no longer
           - open up if the requested sound in not found as a
           - FILE in your SOUNDDIR.
25.12.1996 - Added MCF Configuration File(s) validation to
           - MCF.AMIRX.  It should no longer crap out if one
           - of the configuration files is missing.
           - BTW:  Merry Christmas to all you AmIRC and MCF users.
22.12.1996 - Fixed a bug with DCCGET that crept into MCF with
           - the version 4.10 update.
           - Changed the way CTCP_Sound.AMIRX handles multiple
           - assigned directories when sending thru the GUI feature.
           - Added MCF_Auto.AMIRX.  This is a GUI Front-end for
           - the setting of all current MCF Startup variables.
21.12.1996 - Changed speech to run as background task.
20.12.1996 - Added Validation of MCF.Startup variables which
           - is performed at startup and when %AUTO is done.
18.12.1996 - Added new script: MCF_SAbuser.AMIRX.  Manually
           - exclude a user from making your computer speak.
           - Re-structured MCF.AMIRX for future ease of updates.
           - Added new %AUTO variable PSOUND:  if =  Y  MCF
           - will play soundfiles you have that you send to
           - other users.
14.12.1996 - Compressed Speech coding for greater efficiency.
           - Got sick of UPPER Cased sound filenames.  Changed
           - MCF.AMIRX so that Soundfile name will be in mixed case.
           - Fixed a bug in MCF_SOUND.AMIRX when NOSOUNDS was = 3.
13.12.1996 - Changed the method of scanning for Services on Dalnet
           - so that ChanServ/MemoServ/NickServ won't get ignored.
           - Added Speach to MCF.AMIRX.
11.12.1996 - Fixed a problem in the MCF_HELP.AMIRX script.
24.11.1996 - Added CTCP Functions: MCF_ClientInfo.AMIRX and
           - MCF_Finger.AMIRX
09.11.1996 - Fixed problem with Env Clips in MCF.AMIRX.
04.11.1996 - Fixed MCF_SOUND to switch to NOSOUNDS = 2 when requestor
           - is you.  No more   !yournick Sound.wav  sent to channel.
```

```
             - Fixed %RE function to set variables/clips correctly.
             - Added missing functions to MCF:MCF.Cmds.
01.11.1996 - Added coding to MCF to Decode an Encoded Topic upon
             - Joining a channel that has an Encoded Topic.
30.10.1996 - Fixed Flood Protection routine so that it should not
             - ignore users of "Service@Dal.net" (ie: NickServ, ChanServ).
             - Fixed MCF USE/SAVE option to save all options.
28.10.1996 - Removed "SFPROTECT.?" MsgPort.  Deleted MCF__HELP.AMIRX,
             - MCF__STATUS.AMIRX, and MCF_SFPROTECT.AMIRX.
11.10.1996 - Changed Greets so that if Greets was a sound, it won't
             - Echo back to the screen.
             - Corrected a bug in the GUI of CTCP_Sound.AMIRX.  If
             - SoundDir was an assignment and assigned to only 1 dir,
             - the script was dropping first letter of the filename.
10.10.1996 - Added MCF Aliases installation to the Install_MCF script.
             - Added Configuration update and obsolete delete to the
             - Install_MCF script.
09.10.1996 - Made changes to CTCP_Sound.AMIRX so that the sound
             - requests it sends are playable by Win95 machines.
08.10.1996 - Added 2 new options to the %AUTO GREETS setting.
             - Changed Greets to echo greet back to screen.
07.10.1996 - Moved all ENV: config files to MCF:.
06.10.1996 - Fixed MCF_SOUND.AMIRX to drop IBM style drive letters and
             - directory paths from the sound filename on a ctcp sound request.
05.10.1996 - Fixed bug in Un-Ignore processing of the flood protection
             - routine.
             - Fixed flood protection to not count size on valid ctcp
             - NOTICE Replys.
             - Added Greets function.  Greets by Nick or a random greet.
04.10.1996 - Corrected oversite that allowed a text flood from a user
             - who was already ignored for a ctcp flood.
29.09.1996 - Re-Structured the way MCF processes it's Flood Protection
             - information. (Should result in faster response)
             - Fixed a bug so that a user who is auto-ignored, will
             - no longer get any response from MCF.
27.09.1996 - Fixed a bug in the channel scanning procedure of
             - the !Nick Auto-Request function.
23.09.1996 - Tweaked the CTCP channel flood a little more.
22.09.1996 - Fixed protection to ignore TEXT on a channel CTCP flood.
             - Fixed an error in the help feature of DalHelper.amirx.
21.09.1996 - Corrected a minor error in MCF_SOUND.AMIRX Not-Found processing.
20.09.1996 - Added Auto-Get feature for Auto-Requested sound files.
19.09.1996 - Corrected a byte dropping error in the GUI of CTCP_Sound.AMIRX.
17.09.1996 - Compressed all scripts using RexxOpt.
15.09.1996 - Fixed several bugs in the DalHelper.AMIRX script.
             - Changed GUI Multi-dir requestor in CTCP_Sound.AMIRX.
14.09.1996 - Added SOUNDDIR as an MCF Clipboard variable.
             - Added RexxReqTools GUI to CTCP_Sound.AMIRX.
13.09.1996 - Fixed bug in %AUTO USE/SAVE routines in main MCF.AMIRX script.
             - Changed MCF_SOUND.AMIRX to request a not-found soundfile
             - only once.
06.09.1996 - Fixed bug in decryption routine in main MCF.AMIRX script.
05.09.1996 - Added more features to DalHelper.AMIRX. (More to come)
01.09.1996 - Added List function to MCF_GETSOUNDS.AMIRX.
             - Added /Topic (in place of /nick) option to MCF_ENCODE.AMIRX.
             - Added DalHelper installation procedure to Install_MCF.
29.08.1996 - Corrected error in Join script. Tried to OP when NOT OP.
```

```
                 - Re-coded port waiting procedure because command  WaitForPort
                 - had a tendency of getting lost and returning an RC of 10.
27.08.1996 - Added a couple new features to DalHelper.amirx.
                 - Added Auto-rejoin into DalHelper when MKick or UNban are done.
24.08.1996 - Busy day!  Correct a link error in the Guide file to
                 - the "Pre-Install Testing" node.
                 - Added a Nick change Kicker into the Mass Kick Ban function.
                 - Added "DalHelper.AMIRX" for all people who make use
                 - of either ChanServ or NickServ on Dalnet.
17.08.1996 - Hard-coded "Rexx:" in scripts so that is may work
                 - better on a standard(non-standard) environment.
13.08.1996 - Updated docs with version 4.2 info.
10.08.1996 - Aminet Officially Removed MCF4AmIRC from their Archive
                 - due to the fact that somebody complained, saying that
                 - MCF_MKB.AMIRX (%MKB) is a "War script" and grounds for
                 - removal from the Aminet Archive.
04.08.1996 - Added new %AUTO features.  USE and SAVE.  USE will save
                 - your current MCF config to ENV:MCF.StartUp. SAVE will
                 - do likewise and also save it to ENVARC:
03.08.1996 - Added 2 new remote commands: MCF_OK.AMIRX and MCF_OKB.AMIRX.
29.07.1996 - Re-structured MCF to process a smaller loop on non-valid
                 - hits. Removed the Encryption schemes from both MCF.AMIRX
                 - and MCF_Encode.AMIRX and moved it into a seperate file
                 - in MCF: called "MCF.Encrypt".
                 - Added new Auto-Join on INVITE feature and AUTO variable.
27.07.1996 - Added extra options to NOSOUNDS so the MCF will not
                 - attempt to play the soundfile if NOSOUNDS is greater than 5.
25.07.1996 - Changed some internal coding to use "RAW" instead of "SAY".
23.07.1996 - Added 10 more encryption schemes for use in encoding
                 - messages.
21.07.1996 - Added support for decoding encrypted messages
                 - sent with version 1.3.3 of Cybeonix ENCODE function.
                 - Added Cybeonix compatible encryption routine
                 - for sending encrypted messages.
19.07.1996 - Removed un-needed addlib code from the protection
                 - routines.  It is now added in MCF before running
                 - either of the protection routines.
18.07.1996 - Altered interpretation coding in most of the scripts
                 - as partial solution to interface linkage problem.
                 - Added internal subroutine "FindPort" for when MCF
                 - starts and can't find the AMIRC ARexx MsgPort.
17.07.1996 - Discovered what the major problem was of some users
                 - who have been having trouble trying to run MCF.
14.07.1996 - Added new Extra command for sending a sound to a
                   channel as you join it.
                   Added the use of the server Silence command to the
                   flood protection routine.
                   Corrected an error in MCF.AMIRX and MCF_JOIN.AMIRX
                   Channels listed did not always match up with their
                   count.
13.07.1996 - Altered the parsing of SOUND requests to make it more
                   compatible with Sounds sent by other platforms (IBM).
11.07.1996 - Added support for the auto-fetching of not-found
                   soundfiles in the following format:  !Nick file.wav
                   Added new option (4) to auto variable "NOSOUNDS"
                   Added support for individualized remote comands
                   by inserting the users Nick between the % and command.
```

```
03.07.1996 - Added size checking into the Flood Protection
             routine to provide better protection.  I also
             de-activated the scanning for TEXT flood.
             If you find you need this feature, please contact me.
01.07.1996 - Added a new script and 2 icons to allow a user
             to test-run MCF from the de-arced archive without
             installing it to your Hard-drive first.
21.06.1996 - Corrected termination problem of new SFProtect script.
20.06.1996 - Created 3 new scripts to sequentially handle the
             %STATUS and %HELP commands.  Renamed the real
             script files, created SFProtect.AMIRX which runs
             parallel to MCF.  The new Status and Help scripts
             now do nothing more than stack thier request with
             SFProtect.AMIRX and wait thier turn.
19.06.1996 - Encountered problem with %STATUS and %HELP when
             run as background tasks.
17.06.1996 - Re-Coded MCF and all remote command scripts to run
             as background tasks instead of running under MCF.
15.06.1996 - Added a new variable name and value to MCF.StartUp.
             NOSOUNDS  0-3   Alters MCF's response to a Soundfile
             not found conditions from a CTCP SOUND event.
             Added a new Auto-response function as option 3 for
             NOSOUNDS.  It will issue a  CTCP GETSOUND Soundfile
             to the person sending you the CTCP SOUND.
14.06.1996 - Turned CTCP events to possible functions and added
             valid CTCP functions to the MCF.Cmds file.
12.06.1996 - Added new level 99 command:  %OPTS.
09.06.1996 - Corrected problem in the MCF_MKB.AMIRX script.
02.06.1996 - Added Channelized authorizations and Privledged
             channel settings.  Restricts OPs to only users with
             that channel listed on their MCF.Ops entry.
29.05.1996 - Added new Extra command:  Say.AMIRX.
25.05.1996 - Corrected problem related to cleaning up the Clipboard.
           - Changed the %OP function to work without an argument.
21.05.1996 - Created an Installer script for Installing MCF.
18.05.1996 - Corrected typo in docs.
           - Corrected an error in MCF_MDO.AMIRX.
12.05.1996 - Added TEXT and PRIV scanning/auto-ignore to Protection process.
           - Added Sounds to the CTCP scan of the Protection process.
           - Corrected End of Client error. Now closing Protection MSG Port
             as well as exiting the MCF_PROTECT.AMIRX script.
11.05.1996 - Moved MCF_JOIN.AMIRX to ENV: to reduce HardDrive accesses.
10.05.1996 - Rewrote Flood Protection processing. Now queued with MCF_QProtect.
08.05.1996 - Changed MCF to make use of ENV settings.
06.05.1996 - Corrected a Multi-Client ClipBoard channel problem.
05.05.1996 - Added new function: MCF_Alias to make using level 99 commands easier.
02.05.1996 - Corrected an error that ocurred when RCMDS was set to N.
           - Rewrote the MCF_STATUS function.
02.05.1996 - Added 9 more .help files to the MCF: directory.
01.05.1996 - Changed FLOOD Protection to run as a parallel process.
25.04.1996 - Added OPTS function to display current Options of MCF.
23.04.1996 - Added external command for sending CTCP SOUND to users easily.
20.04.1996 - Added UnIgnore function.
           - Added FLOOD Protection.
20.04.1996 - Added support for processing CTCP Sound commands.
19.04.1996 - Changed MDO function to De-OP users at a rate of 10 per De-OP command ←
   .
```

```
              - Disabled Auto-OP function if OPS not = Y
14.04.1996 - Resolved problems brought about in a Multi-Client environment.
13.04.1996 - Added Remote Commands on/off setting as AUTO switch RCMDS
12.04.1996 - Added ChannelOp recognition character as AUTO stitch OPCHAR
11.04.1996 - Fixed Auto-Start in Multi-Client environments by setting a clip
              variable named after the AmIRC ARexx address() of requesting client.
08.04.1996 - Increased display size of Nick/User name in USERS function.
04.04.1996 - Release of Version 1.0 of MCF
```

1.16   About the author...

```
The Author's Addresses...

Donald T. Becker
335 S Chestnut St
Mount Carmel PA 17851-2219
USA


MCF Support: http://home.ptd.net/~strdustr/MCF.html


E-Mail:      dtbecker@ptd.net

IRC:    StarDustr!StarDustr@*.sun.ptd.net (Space Ranger)
     or StarDustr!StarDustr@*.pcn.ptd.net

        Usually on:   Dalnet  in channel  #AmIRC.



Note:   On IRC, other people use the nick "StarDustr",
        so please verify that it is really me, first.



If you feel the irresistible need to send me money, gifts or an
Amiga 4000 Tower (or maybe a NEW Power Amiga!), I will give you
  a very gracious "Thank You!"


              Only on an


                   /|
                  //|
                 // |
                // .|
               // .|
              // /||
            \\//  /_||
             \/ _/ _||MIGA



P.S.  Some users have mentioned that I should make "MCF"
     shareware and require a registration fee.  I, myself,
     hate paying shareware fees just to get the options of
     the software you want, to work.  If you vountarily
     send any contributions, this will help maintain my
     current attitude about shareware fees and encourage
```

```
me to continue development and FREE distribution of
"MCF".            Thanks to all the users of "MCF"
                  and those who have been a great
                  help in the development and
                  debugging of "MCF".
```