- The lists used by the phonebook and the date panel no longer look quite so weird if using a proportional-spaced font.
- The main screen text snipping scheme has changed a bit and now resembles the standard console device snipping. Also supported is double-clicking on single words in order to snip them.
- The ARexx interface has been entirely rewritten from scratch.

Changes introduced with v3.1:

- Previous releases of the phonebook would not clone single phonebook entries correctly.
- The configuration copying routine will now selectively copy config information to the currently selected phonebook entries.
- The phonebook encryption/decryption routines are a tad faster now.
- The program will now prompt twice for a phonebook access password.
- Comment, phone number, user name and password can be much longer now than they used to be in previous releases.
- The phonebook now only contains the configuration information to differ from the global configuration.
- Previous releases would not save any rate information along with encrypted phonebook files, this has been fixed.
- The configuration copying routine now allows to 'drop' parts of the phone book configuration rather than replacing them with parts of the global configuration. This feature works in conjunction with the new phonebook file format.
- A new feature has been added: as an option, the program will immediately start to redial the currently configured dialing list as soon as the line is hung up or the carrier is lost.
- The program finally includes context-sensitive online help implemented through amigaguide.library (note: Kickstart 3.0 required). Any window to offer a help text will display it when the 'Help' key is pressed.
- The ARexx interface documentation was entirely rewritten. The main program documentation was updated and slightly enhanced. Be sure to reread it!
- Both the text buffer screen and review buffer window now feature new and improved scrollers. Scrolling and screen refreshes are also quite a bit faster now.
- The text searching routine has been enhanced and now works much faster than in previous releases. It also happens to find multiple occurences of a string in a single line.
- Improved text buffer handling, reduced memory fragmentation if running under control of Kickstart 3.0.
- The text buffer will no longer quietly swallow single characters and will finally correctly wrap lines longer than 80 characters.
- The review buffer window now also sports a search function just like the text buffer screen, other useful menu items have been added as well.
- In previous releases the program would miscalculate the number of bytes left on a filing device when receiving a file. It would not take the number of bytes received into account, this has been fixed.
- In order to transfer files one can simply drag the corresponding icons on the 'term' main window (requires that the 'term' window is opened on the Workbench screen).

Changes introduced with v3.3:

- Added the quick dialing menu.
- Enhanced the modem and screen settings
- Fixed a few bugs in the ARexx interface commands `REQUESTFILE` and `REQUESTRESPONSE`
- Unfortunately a single line was missing in the XPR option setup, causing them program not to save the transfer protocol options.
- No longer forces the creation date of an auto-capture file to be included in the file. As an option, will use the naming convention used in previous program releases.
- The program now optionally creates icons for files.
- Added a terminal hex-mode for debugging purposes
- Fonts are now opened using properly initialized DPI and aspect ratio values which results in much better scaled outline font rendering.
- The name of an external emulation library is no longer quietly suppressed.
- It is possible to disable the double-buffered file management routines now.
- Added the text pacing option.
- Apparently, external and internal terminal emulations tried to turn on/off each other's cursors.
- Copying text to the clipboard now permits to append the text to the current clipboard contents.

Changes introduced with v3.2:

- Could not enter phonebook passwords
- The XPR interface now states much more clearly what the likely cause of a serial I/O error might have been
- New sound support functions added
- Added 'OK' and 'ERROR' modem response codes
- The order of messages displayed in the file transfer window has changed (now works top down rather than bottom up).
- The ARexx interface routines did not check whether the serial device handles were available or not (boom!)
- Redid the fast! macro handling
- The text buffer screen did not notice when the buffer contents were cleared
- Retuned the phonebook and dial list routines which could generate Enforcer hits and trash innocent memory

- The standard beep routine would not work.
- Libraries and devices can now be selected using simple menus rather than the file requester.
- The program now flushes capture files to disk every minute in order to insure that at least fragments of the captured session will be safe when disaster strikes.
- Finally got the cross-hatch pattern to draw ghosted list views right.
- The terminal hex mode would write too many bytes per line.
- The program would not pay attention to any icon tool type settings.
- Upon making a connection the dialing routine will no longer drop the line by accident.
- The review buffer window will filter out unprintable characters before displaying text lines.
- The program no longer knocks itself out if an initialization error occurs during the startup phase.
- You can now select the name of the public screen the ‘term’ window is to be opened upon using a menu.
- If using a public screen the program will make use of the corresponding screen font and no longer try to modify it in any weird way.
- The ARexx command `READ` did not pay attention to the translation table settings.
- A configuration file to be loaded upon startup as specified using tooltypes or command line options will no longer be ignored.
- The positions and sizes of certain program windows will be stored in the main configuration file.
- Initially, when a file was received the information window would not display the space left on the output device.
- Introduced the ‘time to connect’ interval.
- Added the serial read quantum.
- The phonebook window will now be opened large enough to hold all its gadgets inside.
- Blinking text display is no longer restricted to eight colour screens, provided the display architecture permits high-resolution screens in more than 4 bit planes.
- The program now exists cleanly if vital resources such as the output screen cannot be opened.
- The file transfer routines should run a tiny little bit faster now.
- Added the transfer performance meter.
- The clear screen control sequence now optionally resets the cursor position.
- Rarely, the text buffer screen would not update the first text line properly. Also, the screen display mode will no longer fall back to HIRES/HIRES_INTERLACED.
- Rarely, the review buffer window would render text in the wrong colours.
- The pen/palette sharing code was submitting incorrect colour codes.
- The review buffer window now responds to the same keypresses as the text buffer screen and the main input window.
- Not all packet menu items were to be invoked using menu shortcuts.

- Rewrote the ARexx/AmigaDOS command execution routines which now execute their commands in truly asynchronous fashion, i.e. once their processes are spawned they keep to themselves and notify the main program when they are finished. This made it possible to remove the odd command handling workarounds and the special ARexx interface code that would take care of ARexx/AmigaDOS command execution.
- The upload queue window code did not get the upload buttons enabled if icons were dropped on it.
- The trap list loading code mixed up the order of commands and sequences, this has been fixed.
- Fixed a monetary quantity formatting bug in the locale support code.
- A malformed text formatting string was causing Enforcer hits in the action logging routine.
- Added a few more 'safety catches' to the screen opening code in order to make sure it does not open screens 'too deep'.
- Changing the screen size in the screen panel did not cause the screen to be reopened, this has been fixed.
- Realized that it was not such a good idea to make most windows simple refresh and changed them to smart refresh. I've probably seen too many Macintosh programs forced to redraw their window contents over and over again.
- The auto-upload panel now sports a button labeled 'Upload from queue' which if enabled will cause the current upload queue to be transferred. Thanks to Bob Maple for pointing me into the right direction.
- Fixed a truly nasty bug in the user interface code: integer type gadgets did not remember their original values when created, they would forget about them when queried and only would report them correctly when the values were changed. There is something I forgot to mention: integer gadgets understand hexadecimal (both 0x.. and $.. notation), binary (%..) and octal (&..) notation. The idea came from Martin Taillefer.
- The date and time entries in log files are now printed in DOS format again.
- Recorded scripts and such no longer include `\*SP` codes instead of plain spaces. This should improve readability.
- With the 'shared screen' feature enabled 'term' would not respect the screen depth settings. This has been fixed.
- The XPR routines no longer queue more than one IORequest, this should help to avoid trouble with some few device drivers.
- The ARexx `QUIT` command really works now.
- Removed the fixed-width font dependencies for Kickstart 2.04. Although windows may look funny if you use a proportional-spaced font, the program should deliver the same functionality as if it were using a fixed-width font.
- Triggering the iconification function with the 'Release serial device when iconified' option enabled will ask you for confirmation if the modem is still online.
- Just for the fun of it added datatypes support. The IFF-ILBM saving routines of picture.datatype are used when saving the window contents to a file.

- The speech volume is given in percent now, but the ARexx interface 'GETATTR' command did not reflect this.
- The sound settings now sport a volume slider which affects all sounds played. Suppose a sound is to be played at maximum volume, i.e. 64 for the current Amiga hardware and the volume slider is set to 25%, then the sound will be played at volume level 16. Setting the volume slider to 0 will cause 'term' not to produce any sound.
- The transfer panel message list dimensions are now auto-adjusted to the screen size. The layout code also tries hard not to make the window overlap the status line.
- Added some bells & whistles to the phonebook window. I hope it still works.
- Moved the user interface code into a shared library in order to make things more complicated.
- The OwnDevUnit feature is no longer linked to the 'Shared access' option.
- All sounds are now replayed in the same fashion as the bell sound, i.e. if a sound is currently being played a request to play another sound will not be satisfied.
- Important file transfer notification messages are now printed in the current highlight colour.
- The destructive backspace mode now offers three choices: off, overstrike and shift. Overstrike mode clears the character below the cursor and shift mode will shift the line contents to follow the cursor to the left.
- The screen panel now features some more options which permit to open the main window on a custom screen as though it were a public screen and an option to split the status line from the main window.
- Tried to squeeze some space out of the control panels with mixed results.
- The text buffer screen no longer uses the main screen display mode by default, it is possible to select the display mode now.
- The built-in ASCII file transfer routines no longer draw upon the current clipboard settings to determine how to send and receive text. There is an all-new preferences editor for this purpose now.
- Added an upload list editor, permitting to collect the files to be transmitted in a list before the transfer is started. You can enter the file names, drop icons on the editor window or on the AppIcon. When you are finished, just press the upload button.
- The packet window string gadget now has room for more than 1000 characters. Note: the 'Load history' command only supports 255 characters per line.
- Added a prescroll option to the emulation settings. The system is not very smart but should get the job done rather quickly; if the cursor is positioned on the last terminal line the number of line feeds in the input data stream is counted. The number of line feeds or the max.prescroll number (whatever is smaller) will determine how many lines to scroll the screen contents up.
- With an empty phonebook loading a new phonebook file would not enable the phonebook list, this has been fixed.
- Added another option to complement the prescroll settings. Testing revealed that the conditions leading to the prescroll feature to be used were met only rarely. The 'max. jump' option will give you roughly the same functionality as the 'max. prescroll' option, the difference is in the handling of the input data

- Clicking on the main window in order to activate it will no longer trigger the character snapping function.
- Rewrote the ARexx 'WAIT' routine to a great deal. It could easily forget to turn serial input processing back on for the main program. After a script would exit you would get stuck with data coming in from the serial line, but none of it would be displayed or worked upon.
- Clicking on the text buffer screen window in order to activate it will no longer trigger the character snapping function. Since there is no safety catch, i.e. the clipboard contents are immediately replaced by what you selected after you let go of the mouse button, this will reduce the chance of losing your current clipboard contents.
- Subtle change in the dialer procedure: if an entry would use the default serial settings they did not replace the current serial settings, even if the previous dial list entry had altered them. However, the original purpose of the default settings was to use the unmodified global settings. The dialer behaviour now respects this, changing the current serial settings back to the global settings, not keeping the changes the previous dial list entry had made.
- In previous program releases trying to make certain phonebook entries not use the default settings was somewhat difficult: you had to change the corresponding settings entries to something different from the global defaults. Things are much easier now, just open the settings editor and click on the 'Use' button.
- The 'Startup/Login macro' has been split into a startup macro and a login macro. The dialing routine will first invoke the login macro and then the startup macro. Only the dialing routine makes use of the login macro.
- Small cosmetic changes to the user interface code: cycle gadgets are a few pixels wider now in order to keep 'CycleToMenu' happy.
- When hanging up the line the logoff macro was never executed.
- Finally discovered why the serial read quantum and all the scheduled events were never processed again once they reported that no further data was available. Now the event response loop updates the signal mask again when it reaches the bottom of the loop to see if any new data came in.
- You can now configure the screen depth, permitting to use the Picasso II chunky display mode with 'term'. Note: asl.library v38-v40 fails to handle nonstandard background pen colours correctly, i.e. the depth slider text may be illegible.
- External emulation libraries, namely xemvt340.library, should work again. I changed the memory allocation call for the XEM support interface, but I have no idea why it did the trick.
- The status window is handled by a coprocess now.
- Some ARexx interface commands now run asynchronously.
- Yet another visual gimmick (sorry, couldn't resist): menu checkmarks and Amiga keys are now scaled according to the current screen display ratio. Note: not really compatible with utilities such as MagicMenu or Silicon Menus. Although the programs will run the menu layout may look odd.
- The phonebook list can be scrolled with cursor keys now. You also get a visual feedback if running under Kickstart 3.x.

- The packet window now supports function keys.
- For technical reasons I dropped the string gadget clipboard support. I suggest that you use StringClip or a similar program instead.
- Clipboard pasting from the review buffer window no longer requires that you activate the main window.
- In the phonebook the edit list (right hand side listview display) indicates whether an entry uses default settings or whether it actually uses custom settings. Custom settings are indicated by asterisks (‘*’).
- The packet window now uses the current terminal text font.
- Pressing Amiga+- in the packet window no longer transmits the current string but changes to the main window while leaving the current string intact.
- Fixed a potential bug in the generic list management module.
- With some file requester patches installed, such as old ReqTools or MFR, the file requester code could fail to notice if a single file was selected in multiselect mode.
- The phone rates are now reported according to the current locale settings, i.e. they take the grouping and special attributes of the local currency into account.
- The IFF-8SVX sound file player code now handles stereo and compressed sound files gracefully (all flavours including uncompressed stereo, compressed stereo, uncompressed mono and compressed mono).
- The ARexx ‘WAIT’ command could fail to report how many characters it pulled from the data stream when a matching string was found. This could result in random characters showing up in the terminal text output.
- The routine to attach the wait mouse pointer to windows and to block input to them was easily losing track of its nesting count. In theory the nesting count could have wrapped around, locking you out. The harmless side-effect was that sometimes windows would not get blocked.
- The ARexx ‘WAIT’ command argument and the wait list may include control sequences now. These are expanded as soon as they go into the list. This feature makes it possible to wait for sequences such as ‘login:\rpassword\r’ but will return result strings which include control characters, so watch out!
- The ARexx ‘READ’ command now allows you to combine the ‘CR’ and ‘NUM’ options. Also, the maximum number of characters to read with the ‘CR’ option is no longer limited to 255 characters.
- The file transfer window will at startup display the name of the currently selected transfer protocol. It displays a default value, leaving the protocol identification to the XPR library.
- The ARexx ‘GETATTR’ command would run into serious trouble if told to put information into a stem variable. The result would be Enforcer hits or crashes.
- If you put ‘term’ into iconified state and press the hotkey combination to bring its screen to the front you will no longer get an Enforcer hit. Instead ‘term’ will exit its iconified state and return to normal action.
- The status line display now coexists much nicer with MagicMenu and the like.
- The status line no longer displays what text mode the terminal window is in (this was rather a silly feature) but rather if the text buffer is currently recording or if it’s frozen.

- If running on the Workbench dropping icons on the ‘term’ window would ask for the type of file transfer (text or binary), but it would get the selection wrong, i.e. if you chose text you got a binary upload and the other way round. This has been fixed.
- When in zoomed state the XPR transfer window will display the name of the file currently being transferred and how much of it has already been transferred (if available). This display will be updated about once a second.
- If icons are to be created for files downloaded it is no longer necessary to turn on file type identification to actually get the icons attached.
- Freezing the text buffer contents now properly updates the text processing routine variables.
- Calling the ‘Print clipboard’ function twice will no longer result in a general system lockup.
- The double-buffered I/O routines now let you configure the buffer size to use. The memory allocation also is a lot more ‘forgiving’ than it used to be: if necessary it will shrink the buffer size until it can allocate enough space.
- To keep naughty applications from switching the cursor key and numeric keymap into applications mode you can lock both key sets now, so they will not to change their current modes.
- The old AmigaGuide release (v34 to be accurate) is supported now, but with limited functionality. The help text is not context sensitive and you need to shut down the AmigaGuide server manually (by closing the AmigaGuide window) if the screen its window resides upon is to be closed.
- ‘RING’ and ‘CONNECT’ messages from the modem are now reported along with the time when they came in.
- The program now complains loudly about outdated catalog files and incorrectly installed locale.library.
- The hotkey settings panel now checks each hotkey description text after it is entered and complains if it is unuseable.
- The ‘QUIET’ command line option (makes ‘term’ start up iconified) no longer crashes if Workbench isn’t running.
- The program now features an all-new user interface.
- There is a new command line option called ‘BEHIND’ which causes the main screen to stay in the background and the main window not to become active upon startup.
- If ‘term’ fails to open a screen and finds out that the requested screen display mode is unavailable it will copy the screen mode the default public screen is in and retry.
- No more trouble with mixed-case device and library names. As you pick them from the list the files are validated, i.e. ‘term’ tries to load the file in question and hunts for the library/device resident tag included. If the tag is found the ‘real’ device/library name is copied from it, replacing the original name the file was opened with. This means that you can select ‘XPRZModem.Library’ using the file requester and ‘term’ will look into the file to find out that the library wants to be opened under the name of ‘xprzmodem.library’. Also included are a type check (i.e. if a library is to be opened only files with a library type resident tag are included in the list) and a brief name comparison (i.e. only

- Creating a new phonebook entry will set the rates accounting data connected with it to zeroes.
- Rewrote the prescrolling/jump scrolling support code which now should get the job more quickly than before.
- Resetting the terminal emulation no longer clears the state of the ‘Wrap cursor moves’ option.
- Rewrote and simplified serial I/O processing, I hope it still works.
- Dialing commands no longer make it into the text buffer.
- The ‘Cancel’ button in the phonebook panel was relabeled, now showing ‘Use’ instead.
- The program no longer puts the upload queue icon into the Workbench window by default, there is a new option to turn it off.
- You can finally edit all the settings to be changed in the phonebook, this includes function keys, cursor keys, translation tables and fast macros. Hold down either shift key to bring up the old file requesters. Note that you will also get the old file requesters if something goes wrong reading and setting up the settings data.
- Added an option to make a hardcopy of the screen contents, invoking the printer graphics dump function.
- All the windows sporting pull-down menus now support menu help, i.e. if you press the help key while a menu item is being selected ‘term’ will bring up the online help page for the corresponding menu.
- Duplicating a phonebook entry now properly duplicates the corresponding rates settings.
- Added new keyboard shortcuts to the phonebook controls. Pressing ‘Del’ untags the currently selected entry, ‘Shift+Del’ untags all entries.
- ‘term’ now looks up the "Fonts" and "Libs" drawers in the current directory and adds them to the "Fonts:" and "Libs:" assignment list. At least for me, this greatly simplifies the installation procedure. Just copy the contents of all distribution archives into a single drawer and let ‘term’ do the rest.
- The fast! macro button list now properly responds to Alt+Amiga key clicks. I also thinned out the code a bit, causing the buttons to render a little faster.
- Relabeled the button in the bottom left corner of the phonebook window again. It now reads "Close".
- The buffer search requesters are now non-modal so you can have them open and continue to use the text buffer display.
- I reworked some parts of the user interface, trying to clarify the functions of menus and buttons. I also removed some redundant button labels and changed all references to ‘directories’ to ‘drawers’.
- Shortened the button labels for the phonebook and the rate panel, so they will finally fit on a 640 x 400 screen using topaz/8. Some button labels now look fairly obscure, sorry about that. I guess I’ll rething the part about the "clarification" again...
- More weird & wonderful changes to the serial device interface code. I hope it still works.
- When hanging up the line using the corresponding menu command the online state is no longer reset to ‘offline’ if in the serial settings the "Check carrier" feature is enabled. This leaves the test for the carrier to the usual routines which will detect if the carrier is really gone.

- Turning off script recording did not reset the program status to 'ready'. This has been fixed.
- Heaven knows why, but v4.0 did not permit changing the screen colours if running in monochrome mode.
- The 'Freeze buffer' menu now properly toggles the state of the capture buffer.
- The terminal settings now take the maximum possible values into account when setting the selection ranges for the number of columns and lines.
- When using a startup script or a startup command the program no longer displays its 'about' window on program startup.
- The ARexx command "GOONLINE" now sets up some more of status variables than it used to do in v4.0.
- Upon startup the carrier signal is checked (provided your serial configuration says that the carrier should be checked) and if it is present the online timer is started.
- Added a bunch more of serial baud rates. Don't overdo it, a standard Amiga won't go faster than 115K baud.
- Replaced the serial rate slider with an integer gadget featuring incrementer arrows. Clicking on the arrows will cycle through all standard baud rates. Note: requires gtlayout.library 1.97 or higher.
- When offline detecting a carrier signal will bring 'term' into online state, provided the 'Check carrier' flag is enabled in the serial settings.
- ^Q now works again.
- The end-of-line character translation scheme was changed into one single unified concept. Both carriage return and line feed characters can now be translated into <cr>, <lf>, <lf><cr>, <cr><lf> or can be ignored.
- The phone rates management is moving from the individual phone book entries into a separate global settings editor. The old rates management style will continue to work, but the new management scheme has priority over it.

  Here is how the new scheme works: you now assign the rates accounting data to area codes rather than to single phone book entries. Suppose you want a special set of rates settings to be used for all phone book entries and phone numbers which start with the area code "009". In this case you would add another group entry, assign a name to it and put "009#?" into the pattern field. The next time 'term' makes a connection to a phone number starting with the digits "009" the corresponding rates settings will be used. 'term' scans the list top-down, so the default settings should be put into the last list entry. The patterns follow the AmigaDOS syntax.
- Finally discovered why so many old phonebook files would cause trouble: the internal conversion routine was *never* called.
- Horrors! The sound.datatype saves invalid sound files with the playback size set to zero, causing 'term' to crash with a 'division by zero' error. The replay routine now handles such odd files.
- The sliders for redial delay and time to connect now finally sport a resolution of a single second rather than ten seconds.

Closer examination has revealed that the interleaved bitmap stuff did not work properly when using Kickstart v2.04. In fact, Kickstart v3.0 is the first operating system release which fully supports interleaved bitmaps for all graphics rendering calls. Previous releases did not take advantage of them, even if set up properly. Sorry folks, you won't be able to use this feature under Kickstart v2.04 any more: I removed the necessary support routines.

- Added pen and text attribute translation. In the emulation settings you will find an option to select nonstandard pens. In this case, these pens refer to the terminal emulation rendering pens and text attributes.
- Tweaked the terminal emulation parser to swallow the Amiga specific commands to turn the cursor on or off (aSCR).
- The device/library selection now also includes ROM-resident modules. At least one multiserial board includes a driver in its ROM rather than on disk. In older releases, this particular driver did not show up in the list, causing users to believe their boards to be damaged.
- Changed the audio channel allocation priority. In previous program releases the channels could be stolen, causing 'term' to hang or crash. Now it's DeliTracker to break down, not 'term' ;-)
- Cloning a phonebook entry did not duplicate the corresponding transfer settings. This has been fixed.
- The serial settings now sport an additional OwnDevUnit control switch. You can now choose to ignore requests to release the serial device driver or to have the device released, causing 'term' to check in intervals of 4 seconds if the device has become available again. The default behaviour (the device driver is released) is still supported.
- There is another sound options, called 'Error sound'. 'term' will play this sound if a certain number of transfer errors have occured. The number of errors to occur can also be set in the transfer settings editor.
- You can now select when the file transfer routines should notify you. You can be notified both at the beginning and the end of the transfer, just at the beginning, just at the end or even never.
- The program no longer reports phone rates after losing a connection if there is no sensible data to report.
- The `"WAIT"` command did apparently pay attention to the case of characters passed in when scanning the wait list for matching entries. This has been fixed.
- For some strange reasons, the VT100 supplementary graphics character set never got loaded. This has been fixed.
- Rewrote the status line display code (yet again). When running on a custom screen you probably won't see any difference, but: open 'term' on a public screen and watch your system performance. No more deadlocks, no more sluggish mouse movements, no more CPU hogging. The display window is a bit larger, but this hopefully won't be a problem. After all, the window mode is usable now. The old BOOPSI code is gone and will probably never return. The new code is in many ways quite a bit nicer than the old code. For example, it is synchronized with the window size changes. As soon as the terminal adapts itself to the new window size, so does the status line display.

as ‘term’ always expected ISO characters to be found in the clipboard (garbage in -> even more garbage out). Nasty, isn’t it? Thanks go to Stephen Bowman for telling me about the problem.

- Under some circumstances the text buffer would get the text font width all wrong, causing characters to be left behind when scrolling the page. This has been fixed.
- Cleaned up gtlayout.library for release, window resizing now works a tad better, although the visual effects are not quite that striking. But then perhaps they are striking, which is why they haven’t returned to work yet.
- ‘term’ also takes care of the screen size now when falling back to a usable screen mode. This should cure the notorious "half height screen" problem.
- The cancel button now does what it should do in the date panel and the modem panel.
- Changed the way how colours are assigned to drawing pens if the selected colours cannot be displayed. The previous method only made sure that there is no black text on a black background, the changes now also take care of white text on white background.
- More changes to the XPR abort handling code; the first request to cancel the transmission while xpr_sread() is being executed now properly follows the rules of how to do things (it stops the read request prematurely and gives the protocol a chance to call xpr_chkabort() and to eventually find out what the user wanted). If you hit cancel again it will – as ‘term’ used to do in previous releases – abort the read request, send a bunch of CAN characters and return with an error. Martin Berndt suggested this.
- I know some of you won’t like it, but the following settings editors are now ‘paged’ to save much space: serial, modem, screen, terminal, emulation, capture and transfer. The nice thing about the new look is that it allows me to save on something else: cryptic abbreviations.
- Discovered some old code left over from prehistoric program releases. The local museum wasn’t interested, so I just discarded it. Unfortunately, the total program size did not drop sharply after I did so.
- Whoops, the sound settings editor did not check for ‘empty’ strings and could tell you that it was unable to locate the file "".
- The picker button of the "Help file" text editing field in the path panel now does what it should do.
- More changes to gtlayout.library, it now respects the window bottom border size gadget and allows the Tab key to be used for cycling through paged settings editors.
- The clipboard and paths settings editors are now paged.
- In paged settings editors, pressing the Tab key cycles through the pages.
- Shortened the english friendly startup reminder message so it fits on NTSC screens.
- When starting up for the first time, ‘term’ no longer complains about missing DSR signals or notifies the user that RTS/CTS handshaking should be enabled. This is done in order to avoid confusion, the reminders and messages will follow later when the user makes the first changes and saves them to disk.
- The XPR transfer window no longer warns about files not fitting on filing system which look suspiciously like ram disks, i.e. are not clearly identified as block mapped filing systems.

- Some of the settings windows are now resizable. Please tell me if this causes any problems. The code is still a bit weird, for example some window sizes can cause the gadgets to overlap the window borders by one or two pixels.
- Small changes to the user interface code. Fixed the notorious double-click bug.
- The AmigaUW terminal resizing code would get invoked before the internal lines/columns variables were set up properly, causing the display to get messed up. This has been fixed.
- When invoking an external file transfer protocol the program now checks if the file name given refers to an ARexx script (it reads the first 256 characters and looks for the comment that identifies an ARexx script), a plain AmigaDOS script (it takes a look at the script file attribute) and eventually accepts the program name as it is. If a file is identified as being a script file it receives special treatment.
- Added another friendly reminder in case the user has enabled the "Connect auto-baud" switch. Apparently, a lot of users have this switch enabled without really knowing what it does and will get into real trouble when making a connection.
- The chat line now gets activated when invoked via menu.
- When failing to allocate enough colours for the terminal window 'term' now falls back to four colour mode.
- Rewrote the dialer (again). If you press skip/abort it will now do what it should do rather than ignoring your commands. The original serial configuration also gets properly restored if the dialer fails to make a connection. Various nice side effects are included. For example, if in waiting state pressing the abort button immediately exits.
- After finding out that the window resizing code did not work properly in programs derived from the review buffer window handling code I gave the original resizing routine another rewrite.
- The ARexx "SEND" command now sports a new "LITERAL" option. With this option the text to send will be transmitted literally, no embedded command sequences or special characters will be evaluated.
- The chat line text entry field no longer filters control characters, even if you enabled this feature in the IControl system preferences editor.
- The text buffer screen could hang when choosing to clear the buffer contents from the menu. This has been fixed.
- The colour remapping that usually took place only in two colour mode (which tries to avoid mapping the same colour to text background and foreground) now also gets applied in four and eight colour modes. Previously, it would ignore the colour mode the user had chosen and just take a look at the depth of the screen the program was using. This could cause all kinds of trouble when running on a 256 colour public screen.
- When running on its own public screen with a window border 'term' could crash if there were still visitor windows open on the public screen.
- 'term' now opens screens as large as possible if the screen settings indicate a specific screen size, but the user has no means to change them. In previous program releases you would get whatever was found in the screen settings, even if you didn't have asl.library v38 or higher handy to change the dimensions.

using an external program no file names will be removed from the ARexx upload list. Here are two examples to get you started:

```
run hydracom device %p speed %b line %c nocarrier rec %> get
```

This will invoke hydracom and start downloading into your download drawer. Put this in to the "Receive" field of your binary transfer settings.

```
run hydracom device %p speed %b line %c nocarrier rec %> send %m
```

This will also invoke hydracom. First you will be asked to select the files to send, then hydracom will transmit them. Put this into the "Send" field of your binary transfer settings.

'term' runs these commands in synchronous fashion, this is why the "run" command is necessary above. Hydracom needs to interface to 'term' while it is running and not currently waiting for the command to complete its task. Aside from the fact that commands are executed in synchronous fashion, they are started just as if you would invoke them using the "Execute AmigaDOS command..." function.

- Fixed a security hole in the review buffer processing code. Previously, the review buffer window could easily lock up when receiving new data.
- Rewrote most of the carrier tracking code. If the carrier is lost during a file transfer 'term' will now properly notice that it is no longer online and run through the usual cleanup procedures.
- More changes to the file transfer settings; for each protocol you use you can now define a specific signature. If 'term' sees this signature in the input data stream it will automatically invoke the protocol in question. The exception is the default protocol which is handled a bit differently. There is no distinction between an upload and a download protocol, since this is how the default protocol works. If the default protocol is an XPR library the library will be open all the time. Whenever the default protocol is invoked, you will be prompted to select the transfer type (text or binary as usual). For auto-activating XPR protocols the signatures will probably be ignored.

  *IMPORTANT: If you are using the Z-Modem auto-upload feature you *MUST* invoke the transfer settings and pick the send signature for the default protocol. Click on the select button at the right side of the "Signature" text entry field. From the list that pops up select "Z-Modem" and save your settings back to disk. If you fail to do so, Z-Modem auto-uploads will *NOT* work.*

  The signatures are scanned in the following order:

  Default protocol (upload) Default protocol (download) ASCII upload ASCII download Text upload Text download Binary upload Binary download

  This means that if you use the same signature for the Default protocol and the Binary upload then the Default protocol will be invoked.

  For now, there are only three signatures built into the program that can be picked from a list: Z-Modem, Hydra and QuickB. QuickB really is not a true signature since it consists only of the ENQ character. Please note that different built-in signature lists will be presented for the upload and download settings.

  For xprzmodem.library it only makes sense to use the upload signature. Starting with v2.0 the library will always filter out the download signature and start the download process all on its own.

- The "READ CR" ARexx command now does again what it should do.
- The code that would cause Enforcer hits within rexxsyslib.library was rewritten to use a different technique to tell free messages and Rexx messages apart.
- The AmigaUW TTY resizing code would crash the machine if the serial device was unavailable. This would happen for example if the serial device driver did not open upon startup.
- The dialing window now displays which dial list entry will be dialed next when in redial delay mode.
- 'term' now properly allocates its work bitmaps when running under Kickstart 3.x, previously it would occasionally fall back to constructing bitmaps on its own which could cause speed penalties.
- Another one bites the dust. Found a really long standing bug in the double-buffered file routines. Can you say buffer trashing, memory losses, crashes? The code used to be very vulnerable to memory shortages. I fixed this and also threw in some extra code to make the buffers quad-longword aligned to help '040 systems with DMA hard disk controllers.
- There is now a bit of new code in the program which opens gtlayout.library. If there still is an old library release in memory it gets flushed first, then the library is reopened. This has the effect of forcing the library to get reloaded from disk.
- Even more changes to the terminal emulation code; previous releases always ignored the current background colour when clearing lines, the screens or moving text around. This has been fixed. Some code also did BitMap peeking which is strictly speaking not allowed. When using fonts with an odd height smooth scrolling could leave pixel trash behind. Some routines, notably those responsible for scrolling and erasing display text, never made sure that the area to scroll/erase was within valid bounds. As the low-level routines always counted upon this data to be correct nasty crashes could result. Some of the new code is far from being highly efficient, but should be much more robust than the old routines. Anyway, those folks looking for a high-speed terminal emulation probably have already chosen a different program.
- The screen settings editor would occasionally assign the wrong colour palette to phone book entry configurations. This has been fixed.
- The routine to reset the text colours to something readable did not take the new emulation pen settings into account. This has been fixed.
- To aid debugging, there is a new switch in the modem settings which tells the dialer to echo commands sent to the modem and to show the modem responses.
- Added another switch to the misc. settings editor. You can now disable those annoying "File ... already exists, do you want to replace it?" requesters.
- Major revamp of the file transfer settings editor. First off, it's no longer that tall. The "Page" cycle gadget cycles through all the individual entries. As there are: the default protocol, the ASCII transfer settings, the text transfer settings and the binary transfer settings. For each protocol you will find another cycle gadget, a text entry field and a big, friendly button labeled "Edit settings...". The cycle gadgets will let you choose between 2..4 possible settings for each protocol. "XPR library" uses the good old XPR interface, the text entry field holds the name of the library to use. "Internal" uses the built-in code. "Default" uses the default protocol. "External program" selects an external program

- Added new "Direct connection" option to the serial settings. If enabled, this switch suppresses RTS/CTS warnings and disables the carrier check. This is useful for direct nullmodem connections and for using the Amiga as a dumb terminal.
- Moved the "Alert" switch from the terminal settings to the miscellaneous settings. This made it necessary to bump the program revision number to v4.4.
- You can now configure the opening size and position of the standard asl requesters (file, font, display mode) in the misc settings. Just select the positioning mode you want (centered or placed relative to the top left corner of the main window) and click on the "Edit..." button. A file requester window will open. Drag it to the position you want and change its size if necessary, then click on "Use". ‘term’ will keep position and size as default values for the requesters to open.
- Rearranged some settings editors, made some room between neighbouring radio buttons and check-boxes, added ‘Tab’ key support to some editors which were missing this feature.
- The notorious screen cloning feature did not work in previous releases, in fact gtlayout.library was broken in many ways and badly needed updating [Klaus Dürr].
- Added LZX archive file type.
- Sorry, no speedbar in this release, I just currently don’t have the time to do it :(
- Added another character send delay option, this time to the modem settings. Some weird east-asian built v.34 compliant modems will happily operate at 57600 baud, but get into real trouble when receiving plain modem dialing and initialization commands. The characters come in just too fast. A little delay between the characters being sent may help.
- The ‘about’ window now sports scrolling credits.
- The phonebook will no longer load empty configuration items (key macros, fast macros, etc.) to edit if the names of the files are not given.
- Due to a bunch of internal changes in order to support scaled characters in more consistent fashion, the option to select half-width characters was removed from the emulation settings. Cut & paste with non-normal sized characters still does not work properly on the screen.
- Redid the logic for the RTS/CTS (DSR check) handshaking mode. ‘term’ now opens the serial device driver with RTS/CTS handshaking disabled, checks the DSR signal and then if necessary closes the device driver and reopens it with RTS/CTS handshaking enabled.
- ‘term’ no longer locks the serial device driver for exclusive access through OwnDevUnit.library when told to open the driver in shared mode.
- Loading the configuration and then choosing "Save configuration" from the main menu will save the current settings under the name of the configuration file loaded last.
- The cursor key control panel now sports a "Default" button.

Changes introduced with v4.3:

frequently cancelled the wrong pages, did not update the colour palette when switching colour modes and moved in utterly strange and mysterious ways when trying to determine whether the user should be allowed to change the colour palette. In a nutshell, your basic waste of code, time & reliability.

- Minor changes to the serial and terminal settings editors.
- Part of the list management was badly broken, this affected in particular the upload list handling (the ARexx upload list, not the upload queue that has its own window). I'm not absolutely certain, but after going through the code for more than three hours I am no longer able to get the machine to crash through the ARexx "SENDFILE" command. It appears that the bug is fixed. Could you give this a try, Chris?
- Caught some more cases in which list and node management routines could corrupt memory.
- The ARexx "SENDFILE" command was sending the upload macro twice on exit.
- Some text buffers handed over to dos.library in order to retrieve file names were not quite as long as 'term' told dos they were. This could cause the usual kind of trouble, but only with very long file names.
- The dialer did not update its OwnDevUnit watcher status when the serial device driver or the ODU options were changed.
- Untag/tag by pattern was still broken, leading to weird select states in the phonebook.
- Swapped the "Use" and "Copy" buttons in the phonebook [Chris Hanson].
- Made the left button row of the phonebook a little smaller by changing the "Exit when finished" checkbox label to "Auto exit". Not that elegant and doesn't even avoid that the phonebook comes up on its own screen with a 640x200 sized display, but at least there is no longer so much of the window hidden from view. The only way to make the window a bit smaller would be to rename "Pattern..." to "Select..." or something, but I cannot do this anymore as the catalogs have been sent to the translators already :(
- The checksumming code did not hit a nul-character, causing the startup to take much longer than necessary. It could also cause 'term' to crash upon startup for no apparent reason.
- Finally (!) managed to fit the phonebook on a 640x200 sized screen. The button size precalculation was using the wrong button data. It did not take the modifications of the phonebook button arrangement into account that took place in the previous betas.
- Squeezed a few more bytes out of the user interface support library code.
- Fixed non-reentrancy problem with the user interface support library.
- Gave 'term' a test on an old A500plus. It runs on an 68k Amiga with 2 MBytes of main memory, but not very comfortably.
- The grouping window in the phonebook did not always close when it had better closed down. Now it does.
- Removed the font scale button from the emulation setings which came back when I restored page #3.

- The XPR init code was throwing the wrong library names in case the setup went wrong. Fixed.
- In the phonebook, clicking on the "rates" settings checkbox now does something sensible. If the button gets ticked, it will receive default rates settings. If the checkmark is cleared, all the rates settings associated with it will get zapped.
- "Dial" is now the default button for the phonebook user interface, I also edited some phonebook gadget labels a bit [Chris Hanson].
- The phonebook window title now keeps track of the number of phonebook entries and those which are tagged [Andreas Kirchwitz].
- The "Select by pattern" code had the selection mode reversed, i.e. "tag matching entries" would untag entries, even if they weren't tagged – with all the usual consequences (can you say "BANG"?).
- Sorting single groups would trash large amounts of memory (Russ was right), as the array to be sorted would hold only a single entry, a memory allocation of the wrong size took place.
- The editing window now responds to the cursor left/right keys, hold down any shift key to move to the beginning or the end of the list.
- "Sort" is now the default button for the phonebook sort panel, this used to be "Sort+Close".
- Rebuilding the main menu did not protect itself well enough against changes in the quick dial menu area, which could cause lots of Enforcer hits and worse.
- The menu builder would depend on the order of phonebook entries to go into groups. It no longer does, I also removed the limitation of the maximum number entries to go into the quick dial menu. Watch out, if 'term' runs out of space it will fall back to the default menu layout without telling you what went wrong.
- The phonebook entry edit window now sports a "Hide" checkbox which is by default checked. If checked, the window won't display or let you edit the user name and password [Russell LeBar].
- Cleared out a lot of dead code, this cut the program size a bit.
- In sixteen colour mode the default text rendering colour is now colour #7, this used to be #15 (i.e. #7 with highlighting). This makes colouring more consistent with the ANSI specs and avoids silly "white on white" rendering.
- 'term' now requires gtlayout.library v24 to run, so make sure you have it installed.
- Replaced all cycle gadgets that were acting as page selectors by the new tab gadgets. I'm still not entirely happy with the visual design of these, point & click is also a bit difficult if the tabs are overlapping one another. The [Tab] key still moves you through the single pages, plus there is a new feature known as "strumming", i.e. you can hold down the mouse button and drag the pointer across all tab tags to see which choices are available. The tags will appear to be plucked out a bit while you review them. When you let go of the mouse button the tag you are currently viewing will become the active tag and the page will be rebuilt. FYI, the tab gadgets will eat a lot of chip memory, please keep an eye on this and let me know if this is a problem for you.
- At the end of an upload through the built-in ASCII transfer routines 'term' would invoke the download macro [Mirko Lukas].

- Better ARexx script recognition within the transfer settings control panel. Previously, in order to have ARexx scripts recognized they had to have their executable file protection bits cleared.
- Changed the way the default buttons look like. Now, what do you think?
- Removed some historic rubble from the screen settings editor which caused screen display modes to be sorted out that would offer less than 640 columns in their default resolution [Russ LeBar].
- Shuffled the controls in the serial and modem settings editor a bit [Russ LeBar].
- New sort algorithm for phonebook and friends.
- All new phonebook. This change is massive and most likely to attract the usual kind of trouble. No documentation exists for this new part of the program yet, sorry. I haven't decided yet how to implement certain features, such as what the checkbox next to the "Rates" settings will do.
- Fixed another load of bugs in the user interface support library.
- If the single character entry window fails to open, the checkmark near the menu item gets cleared.
- The phonebook now uses the brand new popup gadgets.
- Reassigned some keys in the phonebook [Russ LeBar].
- Brought back the v4.3 style xpr_sread routine. Please give this a test and tell me if file transfer reliability improves.
- The phonebook window will no longer expand to display 20 lines of text if there are less than these few entries in the phonebook list [Chris Hanson].
- The phonebook window will grow in the horizontal direction to make more room for the single phonebook entries [Andreas Kirchwitz].
- Just for fun, added support for VT52 escape sequences. Except for "<ESC>^" and "<ESC>_" all should be properly implemented.
- Duplicating an entry in the phonebook would discard special item tracking information in the new entry created, this has been fixed.
- When saving a phonebook file, the currently highlighted group will be saved along with it. The next time the file is loaded, this very group will appear in the listview as the active group again.
- 'term' now reads the system screenmode, serial and font preferences to establish its power-up defaults.
- The "RECEIVEFILE" ARexx command never paid any attention to the file name you could optionally provide, now it does [Stefan Falke].
- If the serial device driver is released on request by OwnDevUnit.library you now get the choice to return to 'term', to iconify 'term' and to quit the program [Christian Hechelmann].
- New options for the phonebook: you can now define which phonebook entries should go into an auto-dial list which 'term' will start dialing right after startup. In addition to that, you can also tell 'term' to keep redialing each entry in the list until it hits the last entry, which will cause it to exit. In order to stop this, either hit cancel in the dialing window, or turn off the "auto exit" feature in the phonebook/clear the dialing list. Last but not least there are new command line options/tooltypes to select a particular phonebook file to use and to turn on the auto dial and auto exit features even if the phonebook was saved without having these enabled [Don Schmelling].

- The windows menu now gets properly updated after the main screen/window is closed and reopened.
- Freezing the buffer via the capture settings and saving them as your default settings never caused 'term' to start up with the buffer frozen, it now does [Geoff Seeley].
- There are now two different ways of getting data into the capture buffer. The old way of doing things (data flow) and a special procedure that stores all the data that gets scrolled or erased from the screen (review). The latter is more faithful to the general concept of a review buffer, but the catch is that you won't see the current contents of the main window. Not all the hooks & lines are in there yet, for example erasing characters in a single line is something the buffer does not track. The code is already in there, but it's commented out. Please let me know if you need it.
- Another addition to the modem settings: if your phonebook entries don't use special modem configurations (in other words, they use the main modem configuration) and you have some modem initialization and cleanup commands in the main modem configuration you do not want the modem to receive while it is dialing, you can now tell 'term' not to send them.
- Mucked around some more with the character raster code that backs cut & paste in the main window. Some optimizations in the code, responsible for clearing the screen when scrolling or erasing text, were not that reliable.
- The dialer was releasing the device driver by request of the OwnDevUnit.library even if it had been configured not to yield it. Even worse, the dialer did not pay attention to whether it should release the device on request while dialing or not [Andreas Kirchwitz].
- Another one bites the dust: the text cut & paste feature supported by the main window no longer gets the terminal emulation into trouble when double-clicking on a word while there is still text selected [Andreas Kirchwitz].
- The transfer panel now finds resident commands given for the external transfer protocol programs to invoke [Andreas Wolff].
- The transfer menu no longer gets ghosted if the default protocol is not an XPR library but an external transfer protocol [Andreas Wolff].
- The 'Upload' buttons found in the upload queue window now cause the current transfer queue to be sent even if it consists of only a single file [Andreas Wolff].
- Changing the default transfer protocol from xpr library to external program can no longer bring the machine down.
- 'term' no longer bangs its head if you accidentally pick an external transfer program and leave the type of transfer protocol set to "XPR library". It now manages to display the error requester without crashing or trashing its own stack [Sven Reger].
- The 'improved' XPR serial I/O code was pulling far too much CPU time. Bob Maple reported about 45%, on my setup it was about 25% which I thought was pretty normal. I went back to the old code and streamlined it a bit, it's now down to about 5%, so the 25% were not that normal after all [Bob Maple].
- 'term' no longer tells you that the function keys, translation tables, cursor keys or hotkeys have been changed on exit if this is not really the case [John Yeung].

'`Show output`'
: If not enabled, this causes the input line not to be echoed in the terminal window.

'`Quit`' Closes the window (corresponds to clicking the close gadget of the window).

Every character entered into this window is shown immediately so that those things where it is better that they should not appear on the screen (like passwords for a mailbox) should be entered in another way.

The contents of every input line are interpreted as a command sequence and therefore can also contain control characters.

If a line taken from the input buffer is sent without change it is *not* stored in the buffer again ('true history' such as known from 'ConMan').

The contents of the input buffer are cleared automatically after the window is closed. *Under no circumstances are the contents maintained until the next call!*

Provided that the packet window is large enough, a list to contain the command line history will be displayed.

'`\n`' Generates a line feed.

'`\p`' Feeds the password of the currently active telephonebook entry into the input stream. *The password is automatically cleared for security reasons when the connection is lost.*

'`\r`' Generates a carriage return.

'`\t`' Generates a tab jump.

'`\u`' Similar to the `\p` command, the `\u` command will feed the current user name into the input stream.

'`\w`'

Depending on how the '`Dial mode`' switch is set in the modem settings, this command either produces `P` for pulse dialing or `T` for touch tone dialing.

'`\x`' Generates a break signal (as with the '`Send break`' menu entry).

'`\^`' Generates a caret character.

'`\~`' Generates a tilde character.

'`\*`' The code to follow the asterisk determines the character to produce. This can be any three digit number or a symbolic name from the following list[1]:

'`NUL`', '`SOH`', '`STX`', '`ETX`', '`EOT`', '`ENQ`', '`ACK`', '`BEL`', '`BS`', '`HT`', '`LF`', '`VT`', '`FF`', '`CR`', '`SO`', '`SI`', '`DLE`', '`DC1`', '`DC2`', '`DC3`', '`DC4`', '`NAK`', '`SYN`', '`ETB`', '`CAN`', '`EM`', '`SUB`', '`ESC`', '`FS`', '`GS`', '`RS`', '`US`', '`SP`', '`DEL`', '`SS2`', '`SS3`', '`DCS`', '`CSI`', '`ST`', '`OSC`', '`PM`', '`APC`', '`NBS`' and '`SHY`'

If none of the mentioned combinations is recognized the character which follows the \ will be fed into the input stream without any changes.

27.2 Caret

This character is used to change the following character to a 'control character'. So the sequence '`^J`' will become a Line feed and '`^I`' becomes a tab jump. The character which follows the ^ has to be located between `@` and `[`, otherwise it is fed into the input stream without changes.

27.3 Tilde

This character causes the program to pause for exactly half a second before it continues to process the following commands.

[1] '`EOU`' may be implemented in a future release

‘Ignore case’

With this switch enabled the search does not distinguish between lower case and upper case characters, i.e. `TEXT` = `Text` = `text`, etc.

‘Only whole words’

If this switch is enabled, ‘term’ will search for whole words only, not for parts of a word. For example, searching for `term` with the ‘Only whole words’ option enabled would stop at the word `term`, but ignore the word `terminal`.

‘Repeat search’

Continues the search process started with ‘Search’. The previously entered search text is carried over.

‘Go to main screen’

Switches to the main screen of ‘term’.

‘Clear buffer’

Clears the contents of the text buffer.

‘Close buffer’

Closes the text buffer screen but leaves the contents unchanged.

of computer) to listen to a sequence of sounds which represent the single digits of the phone number dialled. Pulse dialing involves getting a number of electric pulses, each of which represents a digit of the phone number, transmitted across the line. Tone dialing is usually much faster than pulse dialing, but it isn't supported all over the world. If the receiver of your phone reports a number of beeping sounds when you dial a number you can use tone dialing. If you hear rattling sounds it's probably pulse dialing for you. Let's get back to the '`Dial mode`', if you wish to use pulse dialing, set it to '`Pulse`', otherwise set it to '`Tone`'.

Leave the rest of the modem setup as it is, do not change the '`Connect auto baud`' switch.

3. Next, take a look at the screen settings (see Section 19.3 [Screen panel], page 56). This is where you choose the terminal screen/window look and colours. By default 'term' is configured to open a plain four colour screen using the Amiga default font. This should be sufficient unless you plan to spend most of your modeming time in PC-driven BBSes which keep throwing lots of colours at you.

   Choose how many colours the terminal should use, the switch labeled '`Colour`' will let you choose between '`4 Colours (Amiga)`', '`8 Colours (ANSI)`', '`16 Colours (EGA)`' and '`2 Colours (Monochrome)`'. Each of these settings has a particular default palette attached. The '`Amiga`' mode will use your current system default colours. '`ANSI`' represents the choice of colours the ANSI committee responsible for standardizing a certain terminal command protocol to be the best given the constraints they had. '`EGA`' reflects whatever the engineers who designed the first Enhanced Graphics Adaptor card for the PC considered to be an enhanced colour palette. '`Monochrome`' is my idea how an extremely simplistic, while still readable colour choice could look like. Choose what you find appropriate, but keep in mind that the more colours to use the slower screen updates, scrolling and text output will get. Also, a 16 colour high resolution screen will put your system under additional stress if you are running an older Amiga model which is not equipped with the AGA chip set. Careful please, any changes you make will affect the performance of the program!

   You might want to change the screen mode or the user interface font. When you are satisfied with the setup, return to the main menu.

4. Now it's time to edit the terminal settings (see Section 19.5 [Terminal panel], page 60). This is where you control the basic behaviour of the terminal emulation. If you wish to use an IBM PC style font for the terminal display you can do so by changing the '`Font`' switch to '`IBM PC style`'. Alternatively, you might find it worth changing the '`Text font`' instead which is the font to be used for terminal text output. Note that if the '`Font`' switch is set to anything else but '`Standard`' your '`Text font`' settings will be ignored. Well, actually they will not be entirely ignored, but the IBM PC style font will be opened in the point size you selected.

   Don't touch any other controls, return to the main menu when you are finished.

5. If you are likely to visit a lot of PC BBSes, edit the emulation settings now (see Section 19.6 [Emulation panel], page 63). You might want to turn on the switch labeled '`'CLS' resets cursor position`', otherwise the terminal screen might not get cleared properly when the BBS sends the control codes it considers appropriate for this purpose.

   Leave the rest of the setup as it is and return to the main menu.

```
run hydracom device %p speed %b line %c nocarrier rec %> get
```

The `Hydracom` command must be prefixed with the `Run` command due to the way the protocol interacts with 'term'. For other protocols the `Run` prefix may be omitted.

Now close the window by pressing the `Use` button. Now Hydracom is configured as the binary file transfer protocol. To receive files using the protocol, select the menu item `Download binary file(s)`, to send and receive files at the same time (Hydracom is a bidirectional file transfer protocol) select `Upload binary file(s)`.

If you wish to use the Hydracom signature (see Chapter 21 [Signature panel], page 111 and Section 19.13 [Transfer panel], page 74) to auto-start transmissions, you need to keep a few things in mind. The signature is identical both for uploads and downloads, but using it for both purposes is not a good idea. 'term' will always pick the upload signature first. Hydracom is a bidirectional file transfer protocol which allows you to send and receive files at the same time. This works only when invoking an upload, but not when running a download. If you select a download signature you will lose the bidirectional transfer feature. It is recommended to start transmissions manually.

This setup will always let you transfer data only in one direction. In order to take advantage of the bidirectional transfer feature Hydra offers you will need to make use of two ARexx scripts that should have accompanied 'term'. You only need to modify the commands for `Send` and `Receive` a little:

For `Send` enter:

```
AskUpload.term device %p speed %b line %c nocarrier rec %> send %m
```

And for `Receive` enter:

```
AskDownload.term device %p speed %b line %c nocarrier rec %> get
```

Before the transfer starts you will be asked whether you wish to send and receive data at the same time or whether data should be transmitted only in one direction.

22.4 Escape sequences

When invoking external programs to use for transferring data ‘term’ will build a command line based upon the template given in the transfer settings editor (see Section 19.13 [Transfer panel], page 74). This template can include special tokens, known as escape sequences. Unlike the so-called command sequences (see Chapter 27 [Command sequences], page 129) they are introduced by a percent character (%) and can only be used with external file transfer programs. Please note that you cannot mix command sequences with escape sequences.

The following escape sequences are supported:

‘`%f (Single file name)`’

Inserts a single file name when the program is run. A file requester will open if necessary. If there are still files in the upload queue (see Section 20.15 [File upload panel], page 106) and an upload is to take place the first file name will be inserted and no file requester will appear.

Note: Case matters; %f inserts the file name along with its complete path, %F inserts the plain file name only, omitting the path.

‘`%m (Multiple file names)`’

Inserts a list of file names when the program is run. A file requester will open if necessary. If there are still files in the upload queue (see Section 20.15 [File upload panel], page 106) and an upload is to take place their names will be inserted and no file requester will appear.

Note: Case matters; %m inserts the file names along with their complete paths, %M inserts the plain file names only, omitting their paths.

‘`%p (Port name)`’

Inserts the ARexx port name ‘term’ is currently using.

‘`%d (Device name)`’

Inserts the name of the serial device driver ‘term’ is currently using (see Section 19.1 [Serial panel], page 49).

‘`%u (Unit number)`’

Inserts the unit number of the serial device driver ‘term’ is currently using (see Section 19.1 [Serial panel], page 49).

‘`%< (Source drawer)`’

Inserts the name of the drawer files to be uploaded should be found in. This name will be different for ASCII, text and binary transfers. The default protocol will always use the binary upload path (see Section 19.12 [Path panel], page 73 and Section 19.13 [Transfer panel], page 74).

‘`%> (Destination drawer)`’

Inserts the name of the drawer files to be received should be placed in. This name will be different for ASCII, text and binary transfers. The default protocol will always use the binary

This will invoke the 'Hydracom' program which implements the Hydra protocol which sports bidirectional file transfer and also adds a chat option. This is what the line can expand into when 'term' runs the program:

```
run hydracom device TERM speed 38400 line 14400
   nocarrier rec Work:Downloads get
```

%p expands into the ARexx port name 'term' uses, %b into the baud rate currently used, %c into the baud rate the modem made the connection with and %> into the name of the drawer files received should be placed in.

To complete this example, the following line could be put into the binary Send text entry field:

```
run hydracom device %p speed %b line %c nocarrier rec %> send %m
```

When 'term' runs this program, it will first prompt you to select the files to send, this is what %m does. The files names will then appear in place of the %m characters.

For more information on the escape sequences introduced by the % character, see Section 22.4 [Escape sequences], page 116.

Please note that for 'term' to find the external programs they must either reside in the AmigaDOS Shell search path or need to be prefixed by the complete AmigaDOS path their are located in.

'term' runs the programs in synchronous fashion. Some protocols, such as 'hydracom', however need to be run asynchronously. For such programs it is recommended to prefix the command line with the 'run' command.

22.3 Protocol signatures

Some file transfer protocols sport automatic download and upload functions. At the beginning of a data transmission they send a special data sequence to the remote, indicating that the local side is ready for action. This data is called a signature. With 'term' you can assign a specific signature to each upload and download protocol (see Section 19.13 [Transfer panel], page 74). When 'term' sees this signature in the incoming data stream the corresponding protocol will be invoked.

'Save' Saves the area code & rates accounting information to a file. Upon startup 'term' will read the default area code & rates accounting information from a file named 'rates.prefs', so make sure that your settings file is named correctly for 'term' to find it.

20.17 Parameter panel

When 'term' invokes an external program which is to handle the job of transferring files it can pass special parameters to the program on the command line, such as drawer names. This control panel helps you to build a command line for the program in question.

'Command'
This is where you enter the command to invoke, such as 'run hydracom'.

'1 File' This adds %f to the command line. When the program is invoked a file requester will prompt you to select one single file. Its name will appear in place of the %f characters in the list of arguments passed to the program.

'Files' This adds %m to the command line. When the program is invoked a file requester will prompt you to select a list of files. Their names will appear in place of the %m characters in the list of arguments passed to the program.

'Port' This adds %p to the command line. When the program is invoked the name of the ARexx port 'term' uses will appear in place of the %p characters in the list of arguments passed to the program.

'Device' This adds %d to the command line. When the program is invoked the name of the serial device driver 'term' uses (see Section 19.1 [Serial panel], page 49) will appear in place of the %d characters in the list of arguments passed to the program.

'Unit' This adds %u to the command line. When the program is invoked the unit number of the serial device driver 'term' uses (see Section 19.1 [Serial panel], page 49) will appear in place of the %u characters in the list of arguments passed to the program.

'Source' This adds %< to the command line. When the program is invoked the name of the drawer files to send should be found in (see Section 19.12 [Path panel], page 73) will appear in place of the %< characters in the list of arguments passed to the program.

'Dest.' This adds %> to the command line. When the program is invoked the name of the drawer files should be placed in when received (see Section 19.12 [Path panel], page 73) will appear in place of the %> characters in the list of arguments passed to the program.

'Screen' This adds %s to the command line. When the program is invoked the name of the public screen 'term' uses (see Section 19.3 [Screen panel], page 56) will appear in place of the %s characters

`New` A new trap list entry is added, prompting you to edit it.

`Remove` Removes the currently selected list entry

`Clear` Removes all entries from the list, clearing it.

`Use` Closes the window, using the current trap settings.

`Load` Loads the trap settings from a file.

`Save` Stores the trap settings in a file. *Note: 'term' reads the default settings from the file trap.prefs, so make sure your trap settings are named accordingly if you wish to use them upon startup.*

20.15 File upload panel

'term' permits building a list of files to upload before the upload is started. This list can be built in many ways, such as by dropping the icons of the files to send on the icon labeled '`term Upload queue`', by dropping the icons on the upload panel window, by entering the names of the files in the upload panel window or by using the file requester.

There are two ways to open the file upload panel. You can double-click on the '`term Upload queue`' icon or use the main menu entry '`Upload queue`'. It includes the following controls:

'`Files to upload`'
: This is the list of files to be sent. The text entry field below serves to add new file names or to edit the currently selected file name.

'`Add files`'
: Clicking on this button brings up a file requester to add new files to the list. You can select files from one drawer at a time. The file requester will pop up over and over again asking you to add more files until you press the 'Done' button.

'`Add`' Click on this button to add another file name to the list, you will be prompted to type in its name.

'`Remove`' Press this button to remove the currently selected entry from the list.

'`Clear`' In order to remove all entries from the list, clearing it, press this button.

'`Binary upload`'
: Use this button to upload the listed files in binary mode.

'`Text upload`'
: Press this button to upload the listed files in text mode.

'`Hide`' Click on this button to hide the file upload panel. The list contents will be stored.

'`Redial Delay...`'
Pause until the next cycle through the dialing queue.

Additionally, the following controls are available:

'`Skip`' With this function the current dialing attempt is cancelled and the next number is processed. If no succeeding telephone number exists 'term' waits for the next cycle through the dial queue or until '`Skip call`' is pressed again.

There also is a hotkey combination available to accomplish the same task.

'`Remove`' This button works in part similar to the '`Skip call`' button. Additionally, it removes the current phonebook entry from the dialing list.

'`Go to online`'
If the line is very noisy, the connection to a mailbox may have been made, but the `CONNECT` text may be got lost. Pressing this button will cause 'term' to assume that the modem is in fact online now, start the rates accounting and return you to the main window.

'`Stop dialing`'
Operation of this button exits the dial queue (leaving the the dial queue intact) and ends the dialing process.

'`Start script recording on connection`'
As soon as the connection is establish 'term' will start recording incoming text and your responses to it, thus making it possible to create auto-login scripts and such. For more information on this topic see Chapter 31 [Script recording], page 137.

If a connection is successfully made the corresponding entry in the dial queue will be removed.

Selecting the close gadget will close the window and cause the phone book panel to be reopened.

20.13 Printing panel

This control panel is part of the phonebook. It is opened whenever the '`Print`' button is selected and allows for setting the output options.

'`Output file or device`'
This is where you enter the name of the file or device (such as '`PRT:`') the phonebook printout is to be sent to.

'Append rates'
The rate settings of the selected entry will be appended to the current phonebook entry.

'Cancel' Will abort the selection.

Whenever a rate entry is selected, the corresponding parameters ('Pay/unit' and 'Sec./unit') can be edited. If the entry refers to a certain date or a specific day of week three additional buttons are made available:

'Edit' Just as the labels says, will allow you to modify an entry after it has been created.

'Clone' Will duplicate the current rate entry and append it to the list.

'Remove' Removes an entry from the list.

20.11 Copy panel

This control panel allows you to select which parts of the global configuration to copy into the currently selected phonebook entry.

'To all entries'
The selected parts will be copied to all phonebook entries. If any phonebook entries are selected when this action is to be performed, only the selected entries will be affected.

'Copy' This is where you select from which source the configuration information will be copied:

'Global configuration'
Parts of the currently active global configuration will be copied.

'Defaults'
When going online, instead of overriding the currently active global configuration with the supplied local phonebook configuration the corresponding global configuration will be left unchanged.

'Select all'
Selects all parts.

'Clear all'
Clears the current selection.

'Use' Copies the selected items.

'Cancel' The window is closed, no items are copied.

dial the corresponding phone number. Note: only up to 50 phone numbers can be put into the list.

'Auto dial'
: Marks this entry for auto dialing. For more information on this feature, see Section 20.8 [Phonebook], page 96.

'Hide'
: If this switch is enabled, the contents of the 'User name' and 'Login password' text entry fields below will be obscured. If you wish to edit these, you must disable this switch.

'User name'
'Login password'
: Here you can store information to be used later for login scripts and such.

'Settings'
: This group contains entries which consist of a checkbox and a button. If the button is disabled, the phonebook entry will not use the settings controlled by the button.

This concept is central to how the phonebook works and stores, which is why it needs a bit more explanation.

When you start up 'term', it loads its global configuration data from a certain file. This global configuration normally reflects the most frequently used settings, they do not necessarily need to change between different phonebook entries. For example, if you only use one modem the modem settings will probably be the same for all phonebook entries. If it's the same, disable the checkbox corresponding to the modem settings in every phonebook entry. Repeat for every other settings item that never needs to change. When the dialer eventually makes a connection, it copies all the settings items of the entry it has connected to into the global configuration that have the checkboxes enabled. If you want to use special settings for a phonebook entry, enable the checkbox you want, press the button next to it and edit the settings.

'|<' Moves to the first tagged entry.

'<' Moves to the previous tagged entry.

'>' Moves to the next tagged entry.

'>|' Moves to the last tagged entry.

'Close' Closes the edit window.

20.10 Rate panel

'term' will count the minutes you are online and connected to a BBS as soon as a connection is made through the dialing routine. As soon as the connection is lost or you hang up, 'term' will use the information to be specified in this control panel to calculate the amount of money to be paid for the call.

`Delete` Removes the highlighted entry from the phonebook.

`Group` Brings up the group window:

  `Select group`
  From this list, pick the group you wish to add the tagged entries or the highlighted entry to.

  `Or enter group name`
  Alternatively, you may enter the name of a new group to add the entries to.

  `Use` If there are entries tagged, they will be added to the group you have specified, the group will be created if it does not yet exist. If no entries are tagged, the highlighted entry will be used instead. The group window will be closed and the group button and list will be updated.

  `Cancel` The group window will be closed.

`UnGroup`
The currently displayed group will be deleted. This does not delete the entries in the group.

`Auto dial`
This feature corresponds to the switch of the same name in the phonebook entry editor (for more information see Section 20.9 [Phonebook entry panel], page 99). If enabled, 'term' will put all the entries in the phonebook that have the `Auto dial` option set into a dialing list and immediately start to dial after 'term' starts up.

`Auto exit`
If the `Auto dial` switch is enabled, this switch will make 'term' exit after all the entries in the auto dial list are dialled. To avoid exiting, hit the `Cancel` in the dialer.

`Load...`
Loads the contents of a telephonebook from a file.

`Merge...`
Loads phonebook entries from a file and merges them with the currently loaded phonebook.

`Save...`
Saves the contents of a telephonebook to a file.

`Print...`
This button will cause another control panel window to be opened, see Section 20.13 [Printing panel], page 104 for more information.

`Sort...`
Opens the sort window:

  `Criteria`
  This is where you select by which criteria the visible phonebook entries should be selected. `Name`, `Number` and `Comment` should speak for themselves, `Selection` will sort the entries by the order of tag numbers.

'Terminator character'
: Enter the ASCII code of the terminator character to be used for the 'Ignore data past terminator' feature here.

'Quiet ASCII transfer'
: If this switch is not enabled, the built-in ASCII upload/download routines will display the outgoing/incoming data in the terminal window. This option is to let you watch the progress of the file transfer, so that, for example, if the remote does not respond to the data you send, you may want to stop and restart the upload.

'Strip bit 8'
: If this switch is effect each character received or transmitted by 'term' will have its high-order bit cleared.

With ASCII uploads it is important to make sure that end-of-line characters such as carriage return and line feed are properly set up for the remote. While on the Amiga it is common to end a line of text with a line feed character, most editors and such expect a carriage return character to be transferred. This can easily be arranged by setting the 'Send LF' switch to '<<CR>>'.

20.8 Phonebook

The functions described in the following can be found in the 'Modem' menu and relate to the menu entries 'Phonebook', 'Dial' and 'Redial'.

'term' is equipped with a telephone number management system, the phonebook, which is described in the following lines.

'Group button'
: At the top of the window you will find button which displays the name of the group whose contents are displayed below in the list. By default, you will find the name '<< All entries >>' here, which indicates that not a specific group is displayed, but rather all entries the phonebook consists of. You can switch between groups by pressing the [Tab] key.

'List'
: Below the group button the list of phonebook entries belonging to the group in question are displayed. You can move through the list using the [Cursor up] and [Cursor down] keys. To tag any entries for dialing, hold down and [Shift] key and click on the entries using the mouse, alternatively, highlight the entry, then press the [Space] key. To untag them, repeat the procedure again. A double-click on a single entry will tag the entry and immediately start dialing.

'Comment display'
: For each entry you highlight its associated comment is displayed in the area below the list.

'`Character/line delay`'
: The program will respect the character/line delay values to be set using this control panel.

'`Keyboard delay`'
: The program will send character separated by a delay to be determined by the current system keyboard repeat delay.

Note: the 'echo' text pacing modes are to be used with greate care. Certain online services do not echo characters back to the sender as they run only in half-duplex mode. On the other hand most mailbox programs will not echo certain characters, such as escape codes, etc.

'`Quiet transfer`'
: This switch controls whether incoming text will be displayed in the terminal window. You may want to watch how the remote responds to the data sent/received.

'`Skip current file`'
: Stops sending the current file and proceeds to the next.

'`Stop entire transfer`'
: Stops the ASCII data transfer.

In case a file transfer terminates with an unrecoverable error the file transfer window will stay open until explicitly closed by the user so the transfer error report list can be viewed.

20.7 ASCII-transfer settings

'`Text pacing`'
: The mode to determine how text is sent to the remote:

'`Direct`' Each line will be sent without any delay.

'`Wait for echo`'
: The program will wait for each single character sent to be echoed by the remote.

'`Wait for any echo`'
: The program will wait for the remote to return any character in response to any character sent. Typically, this is the case with password prompts issued by BBSes.

'`Wait for line prompt`'
: The program will wait until the remote sends a certain line prompt text.

'`Character/line delay`'
: The program will respect the character/line delay values to be set using this control panel.

‘Packet delay’
: The delay between two packets being sent.

‘Packet type’
: A short description of the data block type employed for data transfer.

‘Block check type’
: The method employed to verify the integrity of the data blocks being transferred (this usually is a form of cyclic redundancy checking).

‘Block size’
: Size of a data block in bytes.

‘Expected time’
: The time the transfer protocol expects the transfer will take.

‘Elapsed time’
: The time elapsed during transfer.

‘Number of errors’
: The number of errors occured during file transfer.

‘Number of timeouts’
: The number of timeouts occured during file transfer.

If the currently active transfer protocol provides the necessary information, two bars will be displayed at the bottom of the transfer window indicating the amount of transferred data and of time to go before the transfer is finished.

‘term’ knows about the Z-Modem data-inquiry sequence the remote receiver issues when expecting files. If recognized, this sequence will cause ‘term’ to display a requester asking for the type of data upload: text or binary. One could call this feature ‘auto upload’. You also have the opportunity to select ‘Abort’ which will transfer the ZModem abort sequence or to click on the ‘Ignore’ gadget which will plainly ignore the fact that the ZModem inquiry sequence has been recognized. *The Z-Modem abort sequence will also be transferred if you select the ‘Cancel’ button in the file requester to appear after selecting text- or binary-upload.* If the ‘Upload from queue’ option is in effect the contents of the transfer queue will be uploaded.

Some transfer protocols will allow you to enter a default receive path the library is supposed to create files it receives in. On request (see Section 19.11 [Miscellaneous panel], page 71) ‘term’ will ignore these settings and use the settings to be changed in the ‘Settings/Paths’ (see Section 19.12 [Path panel], page 73) menu instead.

Each file that is received and which does not remain empty is examined briefly to find out about the file type. If recognized successfully and the corresponding feature is enabled, a small comment indicating the file type will be attached to the file. ‘term’ currently knows about 83 different file types.

'`File transfer finished' sound`'
The sound to be played when a file transfer is finished successfully.

'`File transfer failed' sound`'
The sound to be played when a file transfer is finished unsuccessfully.

'`Modem 'ring' sound`'
The sound to be played when the modem detects a call by a different modem.

'`Modem 'voice' sound`'
The sound to be played when the modem detects a phone call.

'`Error sound`'
The sound to be played when a number of file transfer errors have occured (see Section 19.13 [Transfer panel], page 74).

'`Volume`' This slider affects the volume of all sounds produced by 'term'. Setting it to zero suppresses sound output.

'`Preload sound files`'
If this switch is enabled 'term' will load all sound files immediately rather than accessing and loading them on demand. This may save access time when a sound is to be played but may eat up precious memory.

'`Load`' Load the sound settings from a file.

'`Save`' Save the sound settings to a file.

'`Use`' Use the current settings.

'`Cancel`' Keep the old settings.

There is no fixed size limit to sound files, the amount of available system memory matters. The sound files may be compressed, mono or stereo files.

As of Workbench 2.04 'term' will only load plain IFF-8SVX format sound files. With Workbench 3.x any sound file can be loaded for which there exists a datatypes class. Please note that due to an operating system bug sound files larger than 102,400 bytes will not play correctly under Workbench 3.0.

20.5 Transfer progress panel

The transfer routines open an information window in which a number of transfer parameters are displayed. Additionally, the file transfer can be aborted by clicking either of the three buttons ('`Stop entire transfer`', '`Skip current file`' or '`Stop transfer batch`'). *For most transfer protocols all buttons have the same effect.* Consult the documentation to see if different levels of abort are supported by your favourite transfer protocol.

'`Close`' Closes the window, but keeps the settings.

'`|<`' Places the currently selected macro at the top of the list.

'`<`' Moves the currently selected macro one entry up.

'`>`' Moves the currently selected macro one entry down.

'`>|`' Places the currently selected macro at the end of the list.

20.2 Hotkey panel

This is where the key sequences used to arrange screens and to execute special functions are to be configured.

'`term screen to front`'
: The keys to press to bring the 'term' screen to the front.

'`Buffer screen to front`'
: The keys to press to bring the screen of the text buffer to the front.

'`Skip dial entry`'
: As an alternative to the '`Skip`' button, pressing these keys will skip a dialing entry if the dialing function is currently active.

'`Stop ARexx command`'
: An ARexx script started from within 'term' can be aborted by pressing these keys. Use this function only if pressing `Control + C` does not stop the program execution.

'`Commodity priority`'
: The commodity priority to assign this task to. You may want to change this value if you have more than one program running which uses the same key sequences as 'term'. The program with the higher commodity priority will receive the keystrokes first.

'`Hotkeys enabled`'
: Whether the hotkeys are enabled or not can be toggled by clicking on this button, or by using the '`Exchange`' program to be found in the '`Tools/Commodities`' drawer.

'`Load`' Loads the hotkey settings from a file.

'`Save`' Saves the hotkey settings to a file.

'`Use`' Use the current settings.

'`Cancel`' Keep the original settings.

'term' will refuse to accept invalid keyword combinations. You will be notified by a brief screen flash/bell signal and the cursor will reappear in the text entry field whose contents are rejected.

'ISO-17-(E).prefs'
Spanish 7 bit (ISO code 17) character set.

'ISO-21-(D).prefs'
German 7 bit (ISO code 21) character set.

'ISO-60-(N).prefs'
Norwegian 7 bit (ISO code 60) character set.

'ISO-61-(N).prefs'
Norwegian 7 bit (ISO code 61) character set.

'ISO-69-(F).prefs'
French 7 bit (ISO code 69) character set.

'PC-8.prefs'
Character translation for standard IBM PC style font. If you wish to use these translation tables, make sure to set the 'Font' type in the terminal panel (see Section 19.5 [Terminal panel], page 60) to 'IBM PC style (raw)'.

Unfortunately, there is no translation available for the Norwegian and Danish variants of the PC-8 character set as I do not yet have a fitting Amiga font available. Similar reasons have yet prevented to implement PC-850 character set support.

19.16 Function key panel

This control panel allows setting user definable texts for all ten function keys. All texts are considered command sequences (see Chapter 27 [Command sequences], page 129), a topic which will be covered later in this document.

'Modifier'
All in all 40 keys may be covered with user defined command sequences (Chapter 27 [Command sequences], page 129). As the Amiga keyboard only has ten function-keys this button switches between the modifier keys (Shift, Control, Alt) which, if pressed in addition which a function key, will execute one of the 40 command sequences.

'Load' Load the function key settings from a file.

'Save' Save the function key settings to a file.

'Use'

'Close' Closes the window, but keeps the settings.

data bytes allowed between ACKs. For example, if you set the Buffer size to 64KB because of your floppy, you should also set the Frame size to 65536 bytes.

'`Error limit:`'

This allows you to set the number of sequential errors which will be required to convince ZModem to abort the transfer. The normal value is 10, meaning that 10 errors must happen in a row with no valid data being transferred in order to cause an abort. This setting is provided for those using XPRZModem with a BBS, who may wish to use a relaxed setting, or those with really lousy phone lines who are desperate and patient enough to want the transfer to continue in spite of horrible performance.

'`Auto-activate mode:`'

'`Y = Auto-activate Yes`'

If the comm program supports the ability, the library will automatically go into receive mode when the start of a ZModem download is detected.

'`N = Auto-activate No`'

Don't try to automatically start downloading, make the user activate it.

'`Delete after sending:`'

'`Y = Delete Yes`'

Delete each file after it has been Successfully sent.

'`N = Delete No`'

Don't delete files after sending them.

'`Keep partial files:`'

'`Y = Keep Yes`'

Keep the fragment of a file received so far if file reception is aborted. This allows you to use the Overwrite Resume option above to pick up where you left off on your next attempt.

'`N = Keep No`'

Delete any partially-received file after an aborted transfer.

'`Send full directory path:`'

'`Send path Yes`'

Send full filenames including drawer path to receiver.

'`Send path No`'

Send only simple filenames, not including drawer path.

'`Default received path:`'

Store all received files in this drawer, if option "Use received path" is not checked. Ignored entry if option '`Use received path`' is checked. The path can be any valid existing drawer, with or without trailing / (e.g. '`df0:`', '`Comm:hold`', etc.).

Additional note for 'term' users: the default received path option is ignored if the '`Override transfer path`' *switch in the miscellaneous panel (see Section 19.11 [Miscellaneous panel], page 71) is enabled.*

‘Edit settings...’

Press this button to edit the settings of an XPR library. For an example of how these settings can look like, see Section 19.14 [XPR options sample], page 80.

‘Use’ Use the current settings.

‘Default (phonebook only)’

Drop the current settings, making a connection to the corresponding phone number will leave the corresponding main configuration entry unchanged.

‘Cancel’ Keep the old settings.

Settings for each transfer library are saved in text files in the ‘ENVARC:’ and ‘ENV:’ drawers.

Leave the default transfer library set to the one you intend to use most.

19.14 XPR options sample

You will find an excerpt of the ‘xprzmodem.doc’ documentation file for the ZModem file transfer protocol below which is the default transfer protocol ‘term’ is shipped with. Please note that other file transfer protocols will sport different options and controls, you should consult the corresponding documentation for more information.

‘Text translation mode:’

‘Y = Text Yes’

If receiving, translate CR/LF pairs or solo CR chars to normal Amiga LF chars. Ignore data past ^Z. If sending, suggests to receiver that they should receive this file in text mode.

‘N = Text No’

Receive file verbatim, without changes. If sending, suggest to receiver that they receive this file verbatim, without translations.

‘? = Text status unknown’

If receiving, use sender’s suggestion as to whether to do end of line translations or not. If sending, tell receiver to use default mode, since we don’t know either.

‘C = Text mode set by Comm program’

The library asks the communications program whether or not to use Text mode for each file. If the communications program does not support the necessary ‘xpr_finfo()’ call, or if the call fails, this option acts like T?. From the user’s point of view, what this option normally does is set the Text mode to match

'Text transfer'

This is where you set up the transfer protocol that is invoked when you select the Upload text file(s), Edit & upload text file and Download text file(s) menu items.

'Type' You can either select XPR library, External program or << Default >> here.

XPR library will use an external transfer protocol library, such as 'xprzmodem.library'. This library will be kept open all the time and may for example handle download session automatically. For more information on how to use XPR libraries see Section 22.1 [Data transfer via XPR library], page 113.

External program will invoke a program when necessary. This program has to temporarily take over serial I/O processing. Special facilities are available to pass parameters such as the name of the device driver 'term' uses to the program. For more information on how to use external programs see Section 22.2 [Data transfer via external program], page 113.

<< Default >> will use the default file transfer protocol.

'Send' This text entry field either holds the name of the XPR library to use or the name of the external program for sending textual data. Clicking on the select button at the right hand side of the text entry field will bring up either a file requester or another control panel to select the program name and to edit the program parameters.

'Receive'

This text entry field either holds the name of the XPR library to use or the name of the external program for receiving textual data.

'Signature'

Many file transfer protocols transmit characteristic data to the remote at the beginning of a transmission. This data is called a signature and when found in the incoming data stream 'term' will automatically invoke the protocol in question. This is particularly useful with external programs. This text entry field holds the signature which will when received start a transfer using the current text transfer protocol. For more information on protocol signatures see Section 22.3 [Protocol signatures], page 114.

'Edit settings...'

Press this button to edit the settings of an XPR library. For an example of how these settings can look like, see Section 19.14 [XPR options sample], page 80.

'Type' You can either select XPR library or External program here.

XPR library will use an external transfer protocol library, such as 'xprzmodem.library'. This library will be kept open all the time and may for example handle download session automatically. For more information on how to use XPR libraries see Section 22.1 [Data transfer via XPR library], page 113.

External program will invoke a program when necessary. This program has to temporarily take over serial I/O processing. Special facilities are available to pass parameters such as the name of the device driver 'term' uses to the program. For more information on how to use external programs see Section 22.2 [Data transfer via external program], page 113.

'Name' This text entry field either holds the name of the XPR library to use or the name of the external program. Clicking on the select button at the right hand side of the text entry field will bring up either a file requester or another control panel to select the program name and to edit the program parameters.

'Send signature'

Many file transfer protocols transmit characteristic data to the remote at the beginning of a transmission. This data is called a signature and when found in the incoming data stream 'term' will automatically invoke the protocol in question. This is particularly useful with external programs. This text entry field holds the signature which will when received start an upload using the current default protocol. For more information on protocol signatures see Section 22.3 [Protocol signatures], page 114.

'Receive signature'

This text entry field holds the signature which will when received start a download using the current default protocol. For more information on protocol signatures see Section 22.3 [Protocol signatures], page 114.

'Edit settings...'

Press this button to edit the settings of an XPR library. For an example of how these settings can look like, see Section 19.14 [XPR options sample], page 80.

'ASCII transfer'

This is where you set up the transfer protocol that is invoked when you select the Upload ASCII file(s) and Download ASCII file(s) menu items.

`Cancel` Keep the old settings.

19.13 Transfer panel

This is where the protocols to be employed for file transfers are to be selected. You will also find a handful of additional options here which one way or the other fit into the cathegory of file transfer related data.

`Override transfer drawer`

Each batch file transfer protocol allows you to specify the name of the drawer to place the files it receives in. By default 'term' will redirect the files to a drawer to be specified in the path panel (see Section 19.12 [Path panel], page 73). If this switch is disabled, the internal settings of the current transfer protocol will be used. This may cause files to appear (or rather disappear) in the wrong drawers.

Note: if you turn on this switch, 'term' will deny the file transfer protocol to delete any files as it may remove them from the wrong drawer.

`Set 'archived' bit`

If enabled, this switch will cause 'term' to mark files sent as `archived`.

`Transfer file icons`

This switch works in conjunction with the drag & upload feature (see Section 20.5 [Transfer progress panel], page 90). By default, 'term' will upload only the files whose icons are dragged on the main window or found in the upload queue. If this switch is in effect the icon files will be transferred as well.

`Mangle filenames for upload`

Certain transfer protocols running under MS-DOS get into serious trouble if told to receive files with names which do not match the local naming scheme (8 characters for the name + `"."` + 3 characters for the extension). For example, in such situations ZModem will restart the file transfer over and over again in a row without getting anywhere. To steer clear of trouble you can turn on the '`Mangle filenames for upload`' switch which will cause the file transfer protocol to report 'condensed' file names to the remote receiver. A special algorithm will shrink the file names to the MS-DOS file name template, clearing potentially dangerous character combinations on the fly. This switch has no effect on external programs. *Note: the algorithm may map two different Amiga file names to the same MS-DOS file name, so watch out!*

`Transfer performance meter`

When a file transfer is running, 'term' may optionally display the file transfer performance in a small resizable window. The lines drawn represent the following information:

- Black line This line displayes the current transfer performance (usually heavily oscillating).

'`Create icons`'

If this switch is in effect 'term' will try to provide icons for all files it receives. The following file types (and the corresponding icon files) are supported:

- Text file icon '`ENV:sys/def_text.info`'
- Sound file icon '`ENV:sys/def_sound.info`'
- Picture file icon '`ENV:sys/def_picture.info`'
- Tool icon '`ENV:sys/def_tool.info`'
- Archive file icon '`ENV:sys/def_archive.info`'
- Preferences file icon '`ENV:sys/def_pref.info`'
- Other file types icon '`ENV:sys/def_project.info`'

Text and pictures saved by the program will also get icons attached.

'`Program priority`'

Use this slider to determine the priority under which the 'term' main process is to operate. Adjusting this value can make 'term' perform more reliably in a system which experiences heavy task loading. It is recommended to experiment with this value until a satisfactory state is found. Setting the program priority too high or too low may affect the performance of coprocess services such as the double-buffered file I/O routines.

'`I/O buffer size`'

This slider controls how much memory the double-buffered file management routines will allocate for each buffer. This means a value of 4096 bytes will result in an allocation of 8192 bytes in total.

'`Alert`' 'term' notifies the user of certain events, such as a connection being established or a file transfer action which has just been finished. This switch allows you to select the type of notification:

'`Bell`' A bell signal will be given.

'`Screen`' The 'term' screen will be brought to the front.

'`Bell & Screen`'

A combination of the two actions above.

'`None`' Nothing will happen.

'`Requester dimensions`'

File, font and screen display mode requesters can be made to appear at specific positions and in specific sizes on the screen. This switch controls how they should appear:

'End' Displays the end of the buffer contents.

'Remember position'
If this switch is in effect, the program will remember the text display position between calls rather than jumping to the top or the end of the text buffer.

'Screen position'
The buffer screen will usually not be quite as wide as the system overscan settings permit. This switch determines the horizontal placement of the screen:

'Left' The screen will be left-edge aligned.

'Centre' The screen will be centred.

'Right' The screen will be right-edge aligned. This will bring the screen depth arrangement gadget in line with the other screens.

'Display mode'
This is where you choose the screen display mode the buffer screen should use.

'Search history size'
The text buffer search function maintains a backlog of all the search text entered. The number of texts to remember, before the oldest is discarded, can be set using the 'Search history size' control. In the search text entry field you can use the Cursor up and Cursor down keys to scroll through the previously entered search strings.

'Use' Use the current settings.

'Default (phonebook only)'
Drop the current settings, making a connection to the corresponding phone number will leave the corresponding main configuration entry unchanged.

'Cancel' Keep the old settings.

19.10 Command panel

Here you will find entries for four command sequences which serve four functions:

'Startup command'
At the beginning of every session with 'term' and after a connection has been established by the dialing routine a command-sequence is executed. Do not use this command for auto-login scripts and such, this is what the 'Login command' is for. Note that the dialing procedure executes the 'Startup command' after the 'Login command'.

'Login command'
This command will be executed immediately after the dialing procedure has established a connection. You should use this command for login scripts and such. Note that the dialing procedure first executes the 'Login command' and then the 'Startup command'.

'`Log actions`'

If enabled will write a protocol of each program action (uploads, downloads, dial attempts, etc.) to a file. Each action is listed along with time and date. Carrier-lost events will also note the approximate cost of the call. The log file created by this function is **not intended** for postprocessing via call-log file analyzers. Use the '`Log calls`' feature for this purpose.

'`Log file`'

The name of the file in which the information on the actions executed by 'term' will be stored if '`Log actions`' is enabled.

'`Log calls`'

If this switch is enabled, 'term' will create call-log files in a format compatible with the '`CallInfo`' program. Sometimes this format is referred to as 'NComm format'.

'`Call log file`'

The name of the file in which the information on calls made by 'term' will be stored if '`Log calls`' is enabled.

'`Connect-auto-capture`'

If enabled will automatically open a capture file after successfully making a connection. Any other already open capture file will be closed before proceeding. The files created will bear the names of the corresponding phonebook entries.

'`Filter enabled`'

If selected, command sequences are filtered out before the incoming characters are captured to disk or printer. This makes good sense with noisy lines generating random characters which might scare your printer. It also produces a text file that is much more readable than with all the control sequence codes cluttering up the text.

'`Convert characters`'

This switch works in conjunction with the '`Filter enabled`' option. When using the '`IBM PC style`' terminal font, 'term' receives characters which normally do not have a place in the standard Amiga character set. If the '`Convert characters`' switch is enabled, these characters will be converted into their Amiga equivalents, if there are any, before they go into the capture buffer. Note that this character conversion is always enabled for text captured to the printer.

'`Creation date`'

By default 'term' will append the date of the call made to the name of the auto-capture file created ('`Add to name`'). Alternatively, 'term' will leave the name untouched and store the creation date within the file ('`Write to file`').

'`File drawer`'

This text gadget contains the path in which the the capture files will be created if '`Connect-auto-capture`' is enabled.

text attribute assignments define how blinking, inverse, underlined and highlighted text should be displayed. This control panel is divided into two parts. One part serves to select the text pens, the other part will let you assign the text attributes. At the left hand side of each control you will find a label which indicates the emulation's default value.

There is a limitation in the number of text colour you can choose. While there is a colour mode which makes use of 16 colours, only a maximum of eight colours can be selected. The reason for this limitation is found in the terminal control commands which allow for only eight colours (0-7). The remaining eight colours (8-15) are selected via a text rendering attribute, known as 'highlight'. This means, if text is to be printed in colour 7 and the highlight text rendering attribute is enabled the text will be printed in colour 15. The 'highlight' text rendering attribute always has a special meaning. In all colour modes except 16 it causes text to be output in boldface.

'`Drawing pens`'
: Here you select the text drawing pen order to use.

'`Attributes`'
: Here you select which text rendering attribute to use instead of the default. You can also choose to disable an attribute.

19.8 Clipboard panel

'`Clipboard unit`'
: The clipboard supports several units (0-255) which can be accessed independently. It can make sense to change this value but generally you will probably leave it as unit '0'.

'`Paste prefix`'
: If enabled, the text to send before the clipboard contents are fed into the input stream, see Chapter 26 [Clipboard], page 127.

'`Paste suffix`'
: If enabled, the text to send after the clipboard contents are fed into the input stream, see Chapter 26 [Clipboard], page 127.

'`Convert <<LF>> to <<CR>>`'
: On the Amiga, new lines end with the LF (line feed) character. To simulate typed text, these characters should be converted to CR (carriage return characters) when pasting the contents of the clipboard. If this switch is enabled, the conversion will take place.

'`Text pacing`'
: The mode to determine how text is sent to the remote:

  '`Direct`' Each line will be sent without any delay.

ANSI compliance, the terminal settings need to be changed to use a 80 columns by 25 lines sized terminal window and the IBM PC style font.

'`Wrap characters`'

This function activates the automatical carriage return function which is triggered as soon as the cursor crosses the right screen margin. To avoid unpleasant side-effects, this switch should be activated all the time.

'`Lock line wrapping`'

If this switch is enabled, any requests to change the end of line text wrapping mode will be rejected.

'`Insert mode`'

Normally, 'term' is in overwrite-mode (characters entered overwrite the contents of the screen). If this gadget is activated, typed characters are inserted by pushing all the characters right of the cursor towards the right margin.

The insert-mode does only work for lines. If characters are pushed out of the screen they cannot be restored.

'`New-line mode`'

This gadget activates a special mode in which some VT100 control sequences cause 'term' to perform a linefeed instead of clearing the screen or other serious changes of the contents of the screen.

'`'CLS' resets cursor position`'

As per the VT100 specs, the control sequence to clear the screen is not to change the current cursor position. However, several applications expect it to be moved to the top left corner of the screen. This button will activate this behaviour.

'`Printer control enabled`'

'term' supports the standard VT220 printer control commands. If you do not want the remote application to play with the printer the corresponding support commands can be disabled with this switch. If disabled 'term' will act like a VT220 terminal with no printer attached.

'`Lock text style`'

If this switch is enabled, any requests to change the text rendering attributes (underlined, highlight, blinking, inverse) will be rejected.

'`Lock text colour`'

If this switch is enabled, any requests to change the text rendering colour will be rejected.

'`Wrap cursor moves`'

According to the VT-100 specifications the cursor movements have to stop at the edges of the screen. In spite of this the cursor may leave these borders, especially in ANSI-mode, and may appear at the other side of the screen. This button activates a more 'tolerant' mode.

'`Lock font mode`'

If this switch is turned on, any terminal commands to change the font scale will be ignored.

'Standard'
: The standard text font selected under the 'Text font' settings in this control panel.

'IBM PC style'
: A font similar to the IBM PC text font will be used. No matter how the translation tables (see Section 19.15 [Translation panel], page 83) are configured, outgoing Amiga characters are translated into PC character values. The terminal window will use the font selected under the 'IBM PC font' settings in this control panel.

'IBM PC style (raw)'
: This selection has very much the same effect as 'IBM PC style' but no character translation is performed. The terminal window will use the font selected under the 'IBM PC font' settings in this control panel.

'Send CR'

'Send LF'

These buttons determine the sequences that are sent to the remote if a carriage return (CR) or line feed (LF) character is to be transmitted. Both characters serve as end-of-line indicators.

'_'

The character is suppressed.

'<<CR>>' A carriage return character is sent.

'<<LF>>' A line feed character is sent.

'<<CR>><<LF>>'
: A sequence of two characters (carriage return followed by line feed) is sent.

'<<LF>><<CR>>'
: A sequence of two characters (line feed followed by carriage return) is sent.

'Receive CR'
'Receive LF'
: These two buttons have largely the same effect as the Send CR/LF buttons, they are different in that they affect the incoming data rather than the data transmitted.

'Use' Use the current settings.

'Default (phonebook only)'
: Drop the current settings, making a connection to the corresponding phone number will leave the corresponding main configuration entry unchanged.

'Cancel' Keep the old settings.

'`Active window title bars`'
: The colour to mark the currently active window and selected buttons and list entries.

'`Active window titles`'
: The colour to use when printing text over active window title bars, selected buttons and list entries.

'`Menu background`'
: The colour to render pull-down menus in.

'`Menu text`'
: The colour to use when printing the menu text.

'`Use standard pens`'
: Reset the current pen selection to default values.

'`Use`' Keep the current settings.

'`Cancel`' Return to previous settings.

19.5 Terminal panel

These settings control the basic behaviour of the terminal emulation 'term' uses. More specific emulation options are available in the emulation window (see Section 19.6 [Emulation panel], page 63).

'`Emulation`'
: This is where you select the terminal emulation. Choose one of the following:

  '`ANSI/VT220`'
  : This emulation is a 'melange' of three terminal command sets which themselves are supersets or subsets of one another. Most of the VT220 command set is supported, including some additions made in the ANSI X3.64 specifications. With VT220 its subsets VT100 and VT102 are supported. However, not all the VT52 commands are supported.

  '`Atomic`' A plain text-only terminal mode which filters out terminal commands and special control characters.

  '`TTY`' Also a text-only terminal mode but which displays all control codes and commands it cannot handle on-screen, great for debugging.

  '`Hex`' Another debugging mode which displays all incoming data in hexadecimal notation. If possible the corresponding glyphs will be displayed as well.

  '`External`'
  : This enables the use of external terminal emulation libraries following the XEM v2.0 specifications. In this mode you need to specify the library to use, otherwise 'term' will return to '`ANSI/VT220`' mode.

'Compact'

A very condensed version of the status line is displayed, only the data is shown but no captions. The data is displayed in the following order:

1. Status
2. Terminal type
3. Transfer protocol
4. Baud rate
5. Serial parameters
6. Time of day
7. Online time

'Online display'

This switch determines what type of information is to be displayed in the bottom right corner of the status display:

'Online time'

The time online

'Online cost'

The amount of money to be paid for the connection

'Time & cost'

Both time and money, the display will toggle between both of them every five seconds.

'Colour' This button determines the colour mode the terminal emulation is going to use. Until now, four modes have been implemented:

'4 Colours (Amiga)'

Four colours, optionally blinking.

'8 Colours (ANSI)'

Eight colours, optionally blinking.

'16 Colours (EGA)'

Sixteen colours, as the EGA-palette, optionally blinking.

'2 Colours (Mono.)'

Monochrome, two colours.

'Blinking'

If selected the VT100 blinking option is enabled. This may require to allocate more colours for a specific colour mode than with blinking disabled, so do not be surprised if display performance suddenly drops like a brick.

'Palette'

These buttons are used to select a colour of the screen palette that is to be changed.

'Red/Green/Blue'

Use these sliders to modify the red, green and blue components of the currently active colour.

‘Time to connect’
: ‘term’ cannot measure the time to pass between the remote modem picking up the line and the local modem sending the `CONNECT` message. This slider allows to set the length of this interval. Upon connection, it will be added to the total online time.

‘Drop DTR on hangup’
: Some modems will track the `data terminal ready` line in order to make sure that the terminal program is listening. Once the line goes back to low potential, these modems will drop the line and hang up. Use this button to make ‘term’ take advantage of this feature.

‘Redial after hanging up’
: If this switch is in effect, ‘term’ will redial all the phone numbers still in the dialing list as soon as the line is hung up or the carrier signal is lost.

‘Connect limit’
: This gauge is to set a certain period of time to be counted after a connection is made. When elapsed, a command sequence (see Chapter 27 [Command sequences], page 129) will be executed as to be set using the `Limit macro` settings. If set to `0:00` this function will be disabled.

‘Limit macro’
: A command sequence to be triggered when the time to be set using the `Connect limit` gauge has elapsed. If no text is entered this function will be disabled.

‘Use’ Use the current settings.

‘Default (phonebook only)’
: Drop the current settings, making a connection to the corresponding phone number will leave the corresponding main configuration entry unchanged.

‘Cancel’ Keep original settings.

19.3 Screen panel

‘Display mode’
: Choose the display mode ‘term’ is to use for the main screen here.

‘Screen font’
: This is where you choose the user interface font ‘term’ is to use for the main screen.

‘Faster layout’
: Activating this switch will slow down display updates and window management. Oh well, not always, there may be a speed increase with some configurations, such as with external emulations activated or when using more than four colours on the screen.

: I suggest to experiment with the effects of this button before actually using it permanently.

‘`'No dialtone' message`’

The message to be returned by the modem in case it does not detect any dialing tone on the phone line.

‘`'Connect' message`’

The message the modem emits after detecting a carrier signal. ‘term’ uses this input to determine successful telephone connection, to reconfigure itself and to start the online timer.

‘`'Voice' message`’

The message ‘term’ is expected to receive if the modem detects a voice call. If in dialing mode, ‘term’ will stop the process. The user will in any case be notified of the event.

‘`'Ring' message`’

The message the modem emits if it receives a call, same effects as with the ‘`Voice`’ message.

‘`'Busy' message`’

The message the modem returns if the number which has just been dialed is busy.

‘`'Ok' message`’

The message the modem returns if a command was successfully executed.

‘`'Error' message`’

The message the modem returns if a command was not to be executed successfully.

‘`Dial command prefix`’

The text to be used to prefix each dialing command. By default ‘term’ uses `ATD\\w` as the prefix. This command contains the special escape sequence `\\w` ‘term’ will expand to either `T` or `P`, depending on what the ‘`Dial mode`’ switch is set to.

‘`Dial command suffix`’

The text to be used to append to each dialing command. This is usually the carriage-return character `\\r`.

‘`Dial mode`’

This switch affects whether dialing commands will use touch tone or pulse dialing. Touch tone dialing usually is quite a bit faster than pulse dialing, but not all phone networks support it. This switch requires that either the dial prefix or dial suffix commands include the `\\W` command sequence (see Chapter 27 [Command sequences], page 129). By default ‘term’ will use pulse dialing.

‘`PBX dial command`’

If you are dialing out through a PBX you may need to dial an extra number and wait a bit before dialing the rest of the phone number. Enter the extra number and any other special modem control characters here. By default ‘term’ will use `0,,,` as the PBX dial command. If enabled, this dial command is put between the ‘`Dial prefix`’, the phone number to dial and the ‘`Dial suffix`’. The default command first dials a 0, then tells the modem to wait a bit (each comma causes the modem to wait for a small amount of time, see your modem manual for more information) for the dial tone to occur and eventually dials the rest of the phone number.

‘`Release device`’
: The driver is released, just as if you would select the main menu item of the same name.

‘`Release device, attempt to reown`’
: The device driver is released, but ‘term’ will retry every four seconds to reopen and thus ‘reown’ it again.

‘`Keep device open`’
: ‘term’ always ignores requests to release the device driver.

‘`Serial device`’
: The name of the serial driver to be used by ‘term’. This is usually ‘`serial.device`’ (‘`modem0.device`’ for the internal Supra modem, ‘`sxbios.device`’ for ASDG’s serial IO card, ‘`vectorser.device`’ for the VectorConnection card, ‘`duart.device`’ for the MultiFaceCard, etc.; consult your hardware manual for more information).

‘`Device unit number`’
: The device unit number of the serial driver selected above. This is usually left ‘0’ but can also be used to address multiple serial IO ports.

‘`Use`’ Accept the current settings.

‘`Default (phonebook only)`’
: Drop the current settings, making a connection to the corresponding phone number will leave the corresponding main configuration entry unchanged.

‘`Cancel`’ Don’t use the current settings.

19.2 Modem panel

This is where modem control text and other related parameters are configured.

‘`Modem initialization command`’
: The text to send to the modem after successful program initialization. This text is optional and thus does not need to be present.

The default modem initialization command ‘term’ uses is `ATE1V1Q0\\r` which will do the following with a modem that complies to the Hayes Standard AT Command Set Specification:

1. [E1] Command state character echo is enabled, so the modem echoes back every character you send to it until it makes a connection.
2. [V1] Result codes are displayed as words, so the modem will respond with messages such as `CONNECT`, `NO CARRIER`, etc. rather than numeric codes.
3. [Q0] Result codes are enabled, so that the modem will send a response to any command it receives.

Some modems will appear to ‘lock up’ when the RTS/CTS handshaking protocol is enabled. This may either be due to a faulty connecting cable (not all cables properly connect the pins required for 7 wire hardware handshaking) or due to configuration problems. Some modems factory settings are incompatible with the way the Amiga handles hardware handshaking. In such a case it is very likely that the modem does not set the so-called DSR signal by default. To change this, first turn off RTS/CTS handshaking, then type the command `AT&S0` and press return, now turn RTS/CTS handshaking on again.

The RTS/CTS handshaking protocol must be used for reliable modem connections using transmission speeds of 4,800 baud and above.

‘`Duplex`’ Determines whether characters are echoed back to the terminal screen or not (full, half = local echo).

‘`Buffer size`’

The number to be specified here allows to set the serial driver I/O buffer size. *Every number you enter here will result in ‘term’ and the serial driver allocating twice the buffer size (this is only a word of warning for those among us who prefer buffer sizes of 256K and up).*

‘`Break length`’

Length of the break signal given in microseconds.

‘`High-speed mode`’

Activates a special mode of the serial driver which is to skip a couple of internal parity and stop bits checks resulting in higher data throughput rate. *If this switch is turned on, the serial parameters will be reset to 8 bits per character, no parity and 1 stop bit.* Do not expect dramatic speed increases.

‘`Strip bit 8`’

If this switch is effect each character received or transmitted by ‘term’ will have its high-order bit cleared.

‘`Shared access`’

This switch allows you to run the serial driver in shared access mode. Not all driver types will allow this to happen.

‘`Handle xON/xOFF internally`’

This switch enables the internal processing of the flow-control characters `xON` (= `Control + S`) and `xOFF` (= `Control + Q`). Whenever you press `Control + S` ‘term’ will change its state to ‘holding’ and stop terminal input and output processing. Press `Control + Q` in order to restart.

‘`Pass xON/xOFF through`’

If this switch is enabled, ‘term’ will pass the `xON/xOFF` characters through to the modem rather than quietly discarding them.

‘`Check carrier`’

‘term’ will recognize the `NO CARRIER` message a modem emits when the carrier line signal drops back to low. For maximum safety, ‘term’ will also check the carrier signal line after

‘Phone number’
: If available, the phone number of the system the modem is currently connected to.

‘Comment’
: If available, the phonebook comment corresponding to the system the modem is currently connected to.

‘User name’
: If available, the user name, as available through the phonebook, corresponding to the system the modem is currently connected to.

‘Screen size’
: The size of the terminal output window in characters (columns and rows).

‘ARexx port name’
: The name of the ARexx host ‘term’ is currently using.

‘Buffer size (bytes)’
: The size of the text buffer (see Chapter 25 [Text buffer], page 125).

‘Free memory (bytes)’
: The amount of free system memory.

‘Review’ A review window is opened which basically displays the same text as the text buffer screen. You can scroll through the text displayed both by mouse (see the right hand side scroller) and by cursor keys.

‘Packet’ Opens an input window in which a single line of characters to be transferred across the serial line can be entered. More on this feature is explained below (see Chapter 29 [Packet window], page 133).

‘Chat line’
: This adds a one-line text entry field just above the status line (or the bottom of the main window, whatever is available at the moment). For more information, see Chapter 30 [Chat line], page 135.

‘Fast! macros’
: Opens or closes the so-called fast macro panel (see Section 20.1 [Fast macro panel], page 87). For more information on fast! macros see Chapter 28 [Fast! macros], page 131.

‘Single character entry’
: This may come in handy for serial debugging and those rare cases when there is no keyboard connected to the Amiga. On a 16 by 16 raster layout you will find every single character of the ISO-8859-1 set the Amiga uses. Click on any one to have it transmitted across the serial line. Hold down the left mouse button and drag the mouse across the matrix to see the character code displayed in the window title bar in various notations. Note that with small fonts the binary notation may be cut off.

'`Reset styles`'
: Resets all character style attributes (bold, blinking, inverse video, underlined, etc.) and sets the text colour to the default pen.

'`Reset terminal`'
: Use this menu item to reset the state of the entire terminal emulation.

18.8 Settings

This is where you configure the standard preferences settings. If you select '`New`' in the phonebook window (see Section 20.8 [Phonebook], page 96) these standard settings will be used. Put in your most commonly used settings here. Change individual entries in the phonebook as needed.

'`Serial`' See Section 19.1 [Serial panel], page 49

'`Modem`' See Section 19.2 [Modem panel], page 52.

'`Screen`' See Section 19.3 [Screen panel], page 56.

'`Terminal`'
: See Section 19.5 [Terminal panel], page 60.

'`Emulation`'
: See Section 19.6 [Emulation panel], page 63.

'`Clipboard`'
: See Section 19.8 [Clipboard panel], page 66.

'`Capture`'
: See Section 19.9 [Capture panel], page 67.

'`Commands`'
: See Section 19.10 [Command panel], page 70.

'`Miscellaneous`'
: See Section 19.11 [Miscellaneous panel], page 71.

'`Paths`' See Section 19.12 [Path panel], page 73.

'`Transfer protocols`'
: See Section 19.13 [Transfer panel], page 74.

'`Default transfer protocol`'
: If this item is selected, a transfer settings editor is displayed. The '`Default transfer library`' in the transfer panel (see Section 19.13 [Transfer panel], page 74) determines the contents of this control panel. If the selected default transfer library does not provide these facilities, a simple text requester will prompt for input. Consult the documentation of the chosen transfer protocol for legal options and the values to which they can be set.

If the serial driver has been released by the ARexx interface and has not been reopened yet, this menu item will do it.

If you are still online, the ‘Redial’ and ‘Dial’ menu entries will be disabled. In order to make another call, hang up the line first.

18.5 Transf. (= Transfer)

This menu provides access to file transfer functions. For more information on this topic, see Section 19.13 [Transfer panel], page 74, Section 19.14 [XPR options sample], page 80, Section 20.5 [Transfer progress panel], page 90, Section 20.6 [ASCII-transfer panel], page 93 and Section 20.7 [ASCII-transfer settings], page 94.

‘Upload ASCII file(s)’

This is a pure ASCII-file upload. It was added to allow poor BBS programs to receive text files.

‘Download ASCII file(s)’

This is a pure ASCII-file download. Refer to ‘Upload ASCII file(s)’ for features/options of this mode.

‘Upload text file(s)’

Sends a file/files to the remote receiver using the current transfer protocol. If possible this command will ‘ask’ the transfer protocol to transmit the file(s) in text mode (whatever that means) which may include CR/LF substitution and other gimmicks. Consult the library documentation to find out if your favourite transfer library supports text mode.

‘Download text file(s)’

Request a file/files using the current transfer protocol. Refer to ‘Upload text file(s)’ for features/options of this mode.

‘Edit & upload text file’

Invokes the currently selected (see Section 19.12 [Path panel], page 73) text editor on a file to be selected using a file requester. ‘term’ pays attention to the ‘EDITOR’ environment variable and will use the program indicated by it.

‘term’ will block and wait until the editor has returned.

After the editor has returned, the user will be asked whether the file edited is to be transferred as plain ASCII or via text upload.

‘Upload binary file(s)’

Send a file/files to the remote receiver using the current transfer protocol. True batch upload is supported both through wildcard expressions (‘#?.txt’ will send all files whose names

18.2 Edit

'`Copy`' In order to transfer any currently marked screen text to the clipboard buffer, select this menu item. Text can be marked by double-clicking the select button while the mouse is over a word or by clicking the select button and dragging the mouse. Holding down either shift key will append the selected text to the current clipboard contents. In any other case the new text will replace the previous contents.

'`Paste`' Pastes the contents of the clipboard at the current cursor position provided that the clipboard contains text data. Hold down either `Shift` key to have 'term' include the '`Paste prefix`' and the '`Paste suffix`' (see Section 19.8 [Clipboard panel], page 66) along with the clipboard contents.

'`Clear`' Any currently marked text will be released as soon as any rendering operations are to be executed in the main window. To release marked text manually, select this menu item.

18.3 Cmds. (= Commands)

'`Execute AmigaDOS command`'

Enter the command you want to execute and its command line arguments here.

'`Execute ARexx command`'

This function calls the ARexx server to execute a script file. If the first input character is a ' or `"` the input will be considered as a small program in a single line. Note that this function will not be available if the ARexx server isn't running.

The ARexx command set supported by 'term' is described in the 'term' ARexx interface documentation.

'`Record script`'

Select this menu item to start/stop script recording. More on script recording can be found under script recording (see Chapter 31 [Script recording], page 137).

'`Record line`'

When in script recording mode this menu item will cause 'term' to temporarily switch into full line recording mode, rather than recording only single keystrokes. Pressing the `shift+return` keys has the same effect as calling the '`Record line`' menu item. More on the topic of script recording can be found under script recording (see Chapter 31 [Script recording], page 137).

'`Edit traps...`'

This brings up the trap list editor (see Section 20.14 [Trap panel], page 105. The list includes control sequences 'term' is to look for in the incoming data stream. When a sequence is found

duart.device from Shell or add the tooltype entry DEVICE=duart.device from Workbench.

'QUIET' If this parameter is present, the program will not start opening a display but rather put an icon on the Workbench backdrop, waiting to be invoked. A double-click will bring it to life. This parameter will be ignored in case the STARTUP-Parameter is used along with it.

'SYNC (Shell only)'

If called from Shell 'term' will detach itself immediately allowing the Shell window to be closed afterwards. This effect can be avoided if SYNC is entered in the command line.

'NEW (Shell only)'

Usually, running 'term' twice will cause the screen of the other program to be popped to the front instead of creating a second 'term' process. To avoid this effect, enter NEW as a calling parameter. If called from Workbench, each program will run as a separate process.

'BEHIND' This option will cause 'term' to open its screen behind all other screens and not to activate its window.

'PHONEBOOK'

Similiar to the SETTINGS option you can specify the name of a phonebook file to load upon startup.

'AUTODIAL'
'AUTOEXIT'

These two options are used in conjunction with the Auto-Dial and Auto-Exit features. Even if the phonebook does not have these options enabled, you can nevertheless tell 'term' to enable them as soon as the phonebook file is loaded. It will then go through the routine setups for auto-dialing. Fore more information on these features see Section 20.8 [Phonebook], page 96.

help text file' settings (see Section 19.12 [Path panel], page 73) to point to PROGDIR:term.guide. Once this has been done, pressing the Help key in any window to support online help will bring up a help window.

Newby, Christian Hechelmann, Dabe Murphy, Daniel M. Makovec, Dean S. Pemberton, Eric W. Sommer, Florian Hinzmann, Frank Duerring, Gary B. Standen, Gregory A. Chance, Holger Heinrich, Holger Lubitz, Hung-Tung Hsu, Jason C. Leach, Jason Soukeras, Jay Grizzard, Joel E. Swan, Jonathan Tew, Juergen Zeschky, Julian Matthew, Kai Iske, Karsten Rother, Kay Gehrke, Keith A Stewart, Keith Christopher, Kenneth Fribert, Klaus Duerr, Leon D. Shaner, Marcel Doering, Mark Constable, Martin Berndt, Matthias Merkel, Matthias Scheler, Matti Rintala, Michael Zielesny, Olaf Peters, Ottmar Roehrig, Peer Hasselmeyer, Peter L. Banville Jr., Piotr Kaminski, Robert L. Shady, Robert Reiswig, Rodney Hester, Russell John LeBar, Sebastian Delmont, Stefan Becker, Stefan Gybas, Stefan Hudson, Stellan Klebom, Steve Corder, Sven Reger, Tony Kirkland, William Michael Mushkin and Yves Perrenoud, took care of testing the program – thank you very, very much!

Error reports concerning the ARexx interface should include a sample script to illustrate what is going wrong.

Some hard disk drive controllers temporarily disable interrupt processing while accesses take place. In such cases I recommend to download and upload from the ram disk or to upgrade the hard disk driver.

16. Even though the speech support feature is enabled, 'term' does not speak a single word.

    With the introduction of Workbench 2.1 Commodore ceased to support the speech synthesizer. If you don't have '`narrator.device`' and '`translator.library`' installed the speech support will not work.

17. When downloading files they don't end up in the right drawer.

    The file transfer protocol usually has it's own opinion on where to place files it receives. This behaviour can be changed by editing the 'term' settings, open the transfer panel (see Section 19.13 [Transfer panel], page 74) and turn on the '`Override transfer path`' switch. Now you can select the names of the directories to store files received in using the path panel (see Section 19.12 [Path panel], page 73).

18. Sometimes 'term' stops processing input and output and just beeps when a key is hit.

    Look at the status line, if it displays '`Holding`' 'term' has received an `xOFF` character. Press `Control + S` to restart.

19. When I moved my configuration files into a different drawer and updated my main configuration the phonebook entries started to 'forget' about their settings files.

    'term' uses the environment variable '`TERMCONFIGPATH`' to locate its configuration files. However, the local phonebook entries may have different search paths set. Check the paths settings to see where they are pointing to.

20. Even though several files are selected for transfer only the very first file is sent.

    Not every file transfer protocol supports batch transfers. There is no way for 'term' to tell whether a protocol supports batch transfers, please consult your protocol documentation for more information.

21. Even though everything is set up correctly no ARexx scripts are executed.

    In order for ARexx to work the '`RexxMast`' program needs to be running. Usually, this program is located in the '`System`' drawer of your system partition. Drag it into the '`WBStartup`' folder in order to use it at system startup time. Also make sure ARexx knows where to find your ARexx scripts. Either give a complete path name or copy your file into the '`REXX:`' drawer. Do not rely upon an `Assign REXX: <drawer name> add` call in your '`S:User-Startup`' file to work, as of this writing ARexx does not support multi-volume assignments.

22. Even though the auto-download feature of the current file transfer protocol is enabled no auto transfer takes place while an ARexx script is running.

    Serial I/O processing only takes place if the main program takes care of terminal output.

23. Running 'term' twice from shell does not cause two 'term' processes to be started, instead only the first program is reactivated.

    By default starting 'term' more than once only brings an already running 'term' process to the front. Use the `NEW` keyword to suppress this feature.

(To be continued)

5. I have saved the phonebook and the configuration files to disk and an older ‘term’ release reports that it cannot read them.

   ‘term’ stores version information with the configuration files it saves. Older ‘term’ releases will refuse to read files created by newer releases. Newer releases will almost always read configuration files by older ‘term’ releases.

6. I upgraded from an older ‘term’ release, but the program refuses to read my configuration and phonebook files.

   The phonebook and configuration file format was changed and greatly enhanced with the introduction of ‘term’ 3.1. Older files need to be converted to the new format, this is what the ‘`UpdateConfig`’ program is for that should be included in the ‘term’ distribution. The conversion is easy, just enter `UpdateConfig <old file name> <new file name>`, the program automatically determines whether it is reading a configuration or a phonebook file.

   Caution: the conversion program cannot read encrypted phonebook files, so they should be saved in unencrypted form first.

7. In some BBSes ANSI graphics and text output starts at the wrong screen position, especially after the screen contents are erased.

   The so-called BBS-ANSI terminal command set treats the ‘clear screen command’ different from the VT100 specs, i.e. it expects the cursor to be reset to the home position. You can enable this feature using the `CLS' resets cursor position` switch which can be found in the emulation panel (see Section 19.6 [Emulation panel], page 63).

8. The text buffer window does not show any special characters, such as accented characters, but only dots (‘.’).

   The text buffer window cannot display characters which fall into the range between code #127 and code #159. In order to show any text at all these codes get replaced by the dot character.

9. When I upgraded from an older program release (1.6 - 2.3) to the new ‘term’ release the program would no longer find all its configuration files.

   In order to annoy you and make things generally irritating some of the configuration files were renamed in v2.4, and some were moved to different directories. While the files used to be present in ‘`ENVARC:term`’ ‘term’ now looks for them in ‘`TERM:config`’. If ‘term’ finds no ‘`TERM:`’ assignments, it will create one. If no ‘`TERM:config`’ drawer can be found, it will also be created. The configuration files have been changed as follows:

   ‘`Preferences.term`’
   ‘`term_preferences.iff`’

   New name is now ‘`term.prefs`’

   ‘`Phonebook.term`’
   ‘`term_phonebook.iff`’

   New name is now ‘`phonebook.prefs`’

Unless requested, no update notifications will be sent via standard mail. I recommend that you include an international reply coupon for each update notification you wish to receive.

The author reserves the right to discontinue development of the ‘term’ program.

- An old Telarium/Trillium game for the C64 on 5.25" floppy disk (such as *The Amazon*, *Rendezvous with Rama*, *Fahrenheit 451*).
- An old Lucasfilm game for the C64 on 5.25" floppy disk (such as *Rescue on Fractalus*, *Ballblazer*). Please send only the PAL versions as the original American program versions were tuned for NTSC machines.
- Scenery disks for the SubLogic Flight-Simulator II – except for *Western European Tour*, *Hawaiian Scenery Adventure*, *Japan*, *USA #7*, *USA #9*, *USA #11* and *USA #14*.
- The films *City Slickers*, *Jabberwocky*, *Brazil*, *The Fisher King*, *Time Bandits*, *Annie Hall*, *Zelig* or *Alien* on a PAL-VHS video cassette
- A CD by the Beatles (specifically, *Please please me*, *With the Beatles* aka *Meet the Beatles*, *Beatles for sale* or *Past Masters Vol. 1*), Little Feat (-1989), Weather Report, Paul Simon (1971-1985), Eric Clapton (-1985), Peter Gabriel (1977-1989), Van Morrison or Daniel Lanois
- A book by Michael Crichton, Bruce Chatwin, Raymond Chandler, Terry Pratchett or Steven Meretzky
- Cash and checks (no credit cards – sorry) are always welcome.

Send your contribution to the following address:

Olaf Barthel
Brabeckstrasse 35
D-30559 Hannover
Federal Republic of Germany

If you wish to be notified when program updates become available or wish to order the next update as it becomes available you must include enough money to cover my expenses, see the chapter entitled orders (see Chapter 4 [Orders], page 7) for more information.

No guarantee of any kind is given that the program described in this document are 100% reliable. You are using this material on your own risk.

The program 'term' and the data received/sent by it must not be used for the following purposes:

1. The construction, development, production or testing of weapons or weapon systems of any kind.
2. The construction, development, production or use of plants/installations which include the processing of radioactive/fissionable material.
3. The training of persons to deal with the abovesaid actions.

Listen to your conscience.