

ASC_SEND

Lunqual and mab

COLLABORATORS

	TITLE : ASC_SEND		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY	Lunqual and mab	November 24, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	ASC_SEND	1
1.1	ASC_SEND_english.guide	1
1.2	Disclaimer	1
1.3	Overview	2
1.4	System requirements	2
1.5	Installation	2
1.6	Usage	2
1.7	hotkey	3
1.8	file	4
1.9	requestkey	4
1.10	reqpath	4
1.11	reqfile	5
1.12	stopkey	5
1.13	delay	5
1.14	delaychar	5
1.15	infokey	5
1.16	popkey	5
1.17	editor	6
1.18	recstart	6
1.19	recstop	6
1.20	recfile	6
1.21	min_delay	6
1.22	h_priority	6
1.23	include_mb	7
1.24	(MUI_ASCSEND	7
1.25	(MUI_ASCSEND	7
1.26	(MUI_ASCSEND	7
1.27	(IREC	7
1.28	InputEvents	8
1.29	mui-ASCII_SEND	8

1.30 cli_ascsend	9
1.31 arexx	9
1.32 info	11
1.33 bugs	11
1.34 mui	11
1.35 Schluss jetzt mit dem Unfug !	12
1.36 Index	12

Chapter 1

ASC_SEND

1.1 ASC_SEND_english.guide

Welcome to ASCII_SEND

In this Distribution (3.3 of 6-10-95) 4 different Programs can be found:

1. Mui_ASCSEND, The whole thing...featuring a nice MUI-GUI,
Arexx-Port and a lot of other goodies
needs MUI
Version: 1.19
2. ASCSEND , Simple version, no GUI, no Arexx,
doesn't need MUI
This is only the Ascii-SENDER
Version 1.14
3. IREC , InputEvent-Recorder part of
Mui_ASCSEND,
doesn't need MUI
Version 1.4
4. CLI_ASCSEND , command-line only version;
Version 1.1

```

~~Disclaimer~~~~~
~~Overview~~~~~
~~System~requirements~~
~~Installation~~~~~
~~Usage~~~~~
~~Info~~~~~
~~Bugs~~~~~
~~About~MUI~~~~~

```

1.2 Disclaimer

ASCII-Send is Freeware, do whatever you like with it...

The author cannot be held liable for the suitability or accuracy
of this manual and/or the program(s) it describes. Any damage

directly or indirectly caused by the use or misuse of this manual and/or the program it describes is the sole responsibility of the user her/him self.

1.3 Overview

ASCII_SEND is a Tool, which can convert any given textfile to a stream of ~InputEvents~. These InputEvents are sent to the active Window or (via the Arexx-Port of MUI_ASCSEND) any window you made the active one.

It is a Commodity, in which you can assign a filename to a (user-defined) hotkey. On pressing this key, the file is converted to Inputevents.

There is also an InputEvent-Recorder (IREC) available, which does this thing the backward way: it converts Keystrokes and Mousebuttons to a textfile.

1.4 System requirements

For all of the programs OS2.04+ is necessary.
MUI_ASCSEND needs at least MUI 2.0 (Library V7+), it is localized under WB2.1+.

1.5 Installation

Just move the executables anywhere you like. It is recommended that you keep the things you need in the wbstartup-drawer.
If you use MUI_ASCSEND you will of course need a properly installed MUI V2.0+. If you want to use the english Version of MUI_ASCSEND copy the file catalogs/english/asccsend.catalog to your LOCALE:catalogs/english/ drawer.

1.6 Usage

all executables (ASCSEND, MUI_ASCSEND und IREC) are to be used with Icon-TOOLTYPES, except for MUI_ASCSEND, which has a GUI. The TOOLTYPES can also be given on a commandline.

CLI_ASCSEND doesn't accept TOOLTYPES, you have to use this program from a shell.

Tooltypes for both ASCII_SEND and MUI_ASCIISEND are :

~~HOTKEY[x]~~

```

~~FILE[x]~~~~
~~REQUESTKEY~
~~REQPATH~~~~
~~REQFILE~~~~
~~STOPKEY~~~~
~~DELAY~~~~~
~~DELAYCHAR~~
~~INFOKEY~~~~

```

Tooltypes only recognized by MUI_ASCSEND and IREC are:

```

~~RECSTARTKEY~~~~~
~~RECSTOPKEY~~~~~
~~RECFILE~~~~~
~~MIN_DELAY~~~~~
~~HANDLER_PRIORITY~
~~INCLUDE_MB~~~~~
~~INCLUDE_DELAY~~~~
~~REC_MAXCHAR~~~~~
~~REC_MAX_ON~~~~~

```

the Tooltype

```

~~WAIT~~~~~

```

is only recognized by IREC and

```

~~CX_POPUP~~~~
~~EDITOR~~~~

```

are for MUI_ASCSEND only.

The textfiles you send may contain other special ~InputEvents~

For details read the MUI-ASCII_SEND"} and
CLI_ASCSEND chapters

You can quit ASCII_SEND by

1. by starting it twice (MUI_ASCIISEND doesn't quit, it opens its
window instead.
2. using Commodities-Exchange->Remove
3. by opening the ~Info-Requester~
and selecting "Quit"

1.7 hotkey

USAGE :

HOTKEY[x]=hotkey

Example:

```
HOTKEY0=control alt 1
```

Defines a hotkey als control-alt-1. To use a hotkey, you have to attach a file (using the ~File-Tooltype) to it.

You can have as many hotkeys as you like, but the numbers have to be succesive.

1.8 file

USAGE :

```
FILE[x]=dateiname
```

Example

```
FILE0=work:test/unfug
```

this file is attached to the ~Hotkey~ 0 and will be sent, when the hotkey is pressed.

You can have as many hotkeys as you like, but the numbers have to be succesive.

1.9 requestkey

```
REQUESTKEY=hotkey
```

This hotkey opens a filerequester. If you chose a file, the window which was active before the requester will be activated. You can supply defaults for path and filename using the REQPATH and REQFILE tooltypes

***** Attention *****

This function does not check if the window to activate exists !! DO NOT CLOSE THIS WINDOW !

1.10 reqpath

```
REQPATH=pfad
```

Default path in the filerequester
see also REQUESTKEY

1.11 reqfile

REQFILE=Dateiname

default filename in the filerequester
see also REQUESTKEY

1.12 stopkey

STOPKEY=hotkey

this hotkey stops transferring a file.

Example: STOPKEY=control alt 3

1.13 delay

DELAY=x

global delay between single lines of a file in 1/50s.
Defaults to 12 (1/4s).
see also the DELAYCHAR tooltype
and the ~\d~Inputevent chapter

1.14 delaychar

DELAYCHAR=x

global delay between single characters of a file in 1/50s.
Defaults to 0 (no delay).
see also the DELAY tooltype
and the ~\d~Inputevent chapter

1.15 infokey

INFOKEY=hotkey

Show the Info-Requester.

1.16 popkey

CX_POPUP=hotkey (MUI_ASCSEND only)

opens the MUI_ASCSEND-window and stops transferring files.

1.17 editor

EDITOR=editor (MUI_ASCSEND only)

Editor to use.

1.18 recstart

RECSTARTKEY=taste (MUI_ASCSEND and IREC only)

starts recording keystrokes
to stop press the RECSTOPKEY
or open the MUI_ASCSEND-window

1.19 recstop

RECSTOPKEY=taste (MUI_ASCSEND and IREC only)

stops the recording of keystrokes.
The keystrokes are then converted into a textfile and written with
RECFILE filename

1.20 recfile

RECFILE=dateiname (MUI_ASCSEND and IREC only)

The textfile into which the recorded keystrokes are written

1.21 min_delay

MIN_DELAY=50 (MUI_ASCSEND and IREC only)

delay in 1/50 secs. If two recorded InputEvents have more
delay, a delay-InputEvent (\dxxxx) is written to the file

defaults to 25 (1/4 sec)

1.22 h_priority

HANDER_PRIORITY=100 (MUI_ASCSEND and IREC only)

Priority of the inputhandler used by the InputEvent-Recorder.
should be greater than 50.
defaults to 60

1.23 include_mb

INCLUDE_MB=YES/NO (MUI_ASCSEND and IREC only)

if YES, Mousebuttons are recorded.

1.24 (MUI_ASCSEND

INCLUDE_DELAY=YES/NO

if YES , delays between keystrokes are written to the output file

1.25 (MUI_ASCSEND

REC_MAXCHAR=x

The InputEvent-Recorder will only recognize x characters. When these characters are read, it stops recording and writes a file. You have to turn this option on using the REC_MAX_ON tooltype (or MUI_ASCSEND function)

1.26 (MUI_ASCSEND

REC_MAX_ON=YES/NO

if YES, only the in REC_MAXCHAR given numbers of InputEvents will be read.

1.27 (IREC

WAIT=[ticks]

can be used on the commandline of IREC (as all other IREC-tooltypes)

If you have set REC_MAX_ON=YES and REC_MAXCHAR > 0 und WAIT > 0, then IREC waits x 1/50sec, starts recording and quits after REC_MAXCHAR characters are read. Then it writes the RECFILE and quits.

1.28 InputEvents

ASCII_SEND recognizes a number of special InputEvents.

1. key descriptions in angle brackets

Example

pressing Control-f1: <control-f1>

2. other events

```
\n    = ENTER
\r    = RETURN
\t    = TAB
\d[x] Delay in 1/50 Sec
```

Example: And now for 2 seconds delay : \d100 and on we go...

```
\l[x] Delay in 1/50 Sec between lines
```

```
\c[x] Delay in 1/50 Sekunden between single characters
```

Example: \c20writing slowly...\c0 and quickly...

HINWEIS: The \c und \l events are only valid during the current file

The delays \d, \c und \l can have up to 4 digits use trailing zeroes (as the InputEvent-Recorder) in doubt

```
\(lmd:x,y) "Left Mouse Button down" at the Position x,y
              of the frontmost screen
```

```
\(lmu:x,y) "Left Mouse Button up" at the Position x,y
```

```
\(rmd:x,y) "Right Mouse Button down" at the Position x,y
```

```
\(rmu:x,y) "Right Mouse Button up" at the Position x,y
```

```
\(mmd:x,y) "Middle Mouse Button down" at the Position x,y
```

```
\(mmu:x,y) "Middle Mouse Button up" at the Position x,y
```

1.29 mui-ASCII_SEND

1. When the control window is open, all hotkeys are inactive
2. when opening the window, transferring files is suspended
3. you can save the settings into the program icon.

these icon-files can be used with IREC, ASCII_SEND and CLI_ASCSEND, too.

4. MUI_ASCSEND has an Arexx-Port

1.30 cli_ascsend

CLI_ASCSEND can only be used from a shell.

USAGE:

ascsend FILE/A,WINDOW/K/M,SCREEN/K,WAIT/K/N,DELAY/K/N,DELAYCHAR/K/N

FILE/A textfile to send

WINDOW/K/M the window to sent the file to. If it doesn't exist, ascsend quits without sending a file

SCREEN/K Name of a screen. The file ist sent to the first window of this screen.

WAIT/K/N number of ticks (1/50 s) to wait before sending the file.

DELAY can DELAYCHAR be used also.

Example:

```
ascsend test.file WINDOW SAS/C Editor WAIT 100
```

waits 2 seconds, activates a window called "SAS/C Editor", sends the test.file and quits

1.31 arexx

These are the arexx-commands of MUI_ASCSEND

command	Parameter	err
-----	quits MUI_ASCSEND.	----- quit
hide	-----	iconify and make MUI_ASCSEND inactive
show	-----	activate the hidden MUI_ASCSEND
openwindow	-----	open the control window
closewindow	-----	close the control window

stopkey	-----	stop sending a file
infokey	-----	show the info-requestert
requestkey	-----	show the filerequester NOTE: this call is ignored, when the control window is currently open
send	FILE/A	send a file NOTE: If MUI_ASCSEND is currently sending a file, this call is ignored
string	STRING/A/M	send the string instead of a file
activate_window the given	WINDOW/A	activate the window with title. if the window is not found RC_ERROR is returned. activate_screen SCREEN/A
activate the first window of the		given sreen.
setkey	FILE/A,KEY/M/A	make a hotkey together with a file
start recording.		start_rec ----- NOTE: this call is ignored, when the control window is currently open
stop_rec	-----	stop recording.
recfile	FILE/A	the filename for the InputEvent- recorder.
rec_options	MIN_DELAY/N,PRI/N,MOUSEBUTTONS/S,DELAYS/S,MAXCHAR/N,ON/S	Options of the InputEvent-Recorders Example: rec_options 'MIN_DELAY=30 PRI=90 DELAYS MOUSEBUTTONS'
remove a hotkey-file-pair from the	remove KEY/M/A	list. Example: remove 'control alt x' removes the corresponding key.
-----	remove_all -	clear the entire hotkey-file-list
set		
POPKEY/S,REQUESTKEY/S,INFOKEY/S,STOPKEY/S,RECSTARTKEY/S,RECSTOPKEY/S,KEY/M/A		set the keys Example:

```

                                set POPKEY 'alt f10' sets
                                CX_POPKEY to  "alt f10"
                                delay                                TICKS/N/A
delay between lines

delaychar                        TICKS/N/A      delay between characters
                                reset                                -----
clear all settings and load them

                                new from the icon.

saveicon                        -----          write an icon with the current
settings
```

1.32 info

ASC_SEND was compiled using SAS/C 6.51 under OS 3.0 on an A4000/40/10MB.

Bugreports, flames, gifts and all that stuff to

Snailmail:

Karlheinz Klingbeil
Elzstrasse 42

D-79261 Gutach
Federal Republic of Germany

E-Mail:

lunqual@crazy.freinet.de
lunqual%rab@wsb.freinet.de

1.33 bugs

You'll find bugs.... ;))

1.34 mui

This application uses

MUI - MagicUserInterface

(c) Copyright 1993/94 by Stefan Stuntz

MUI is a system to generate and maintain graphical user interfaces. With the aid of a preferences program, the user of an application has the ability to customize the outfit according to his personal taste.

MUI is distributed as shareware. To obtain a complete package containing lots of examples and more information about registration please look for a file called "muiXXusr.lha" (XX means the latest version number) on your local bulletin boards or on public domain disks.

If you want to register directly, feel free to send

DM 30.- or US\$ 20.-

to

Stefan Stuntz
Eduard-Spranger-Straße 7
80935 München
GERMANY

1.35 Schluss jetzt mit dem Unfug !

Aber genau ;) !

1.36 Index

Index of database ASC_SEND.guide

Documents

(IREC
(MUI_ASCSEND
(MUI_ASCSEND
(MUI_ASCSEND
arexx
ASC_SEND_english.guide
bugs
cli_ascsend
delay
delaychar
Disclaimer
editor
file
hotkey
h_priority
include_mb
info
infokey
InputEvents
Installation
min_delay
mui
mui-ASCII_SEND
Overview

popkey
recfile
recstart
recstop
reqfile
reqpath
requestkey
Schluss jetzt mit dem Unfug !
stopkey
System requirements
Usage

Buttons

~~About~MUI~
~~Bugs~
~~CX_POPUP~
~~DELAY~
~~DELAYCHAR~
~~Disclaimer~
~~EDITOR~
~~FILE [x] ~
~~HANDLER_PRIORITY~
~~HOTKEY [x] ~
~~INCLUDE_DELAY~
~~INCLUDE_MB~
~~Info~
~~INFOKEY~
~~Installation~
~~MIN_DELAY~
~~Overview~
~~RECFILE~
~~RECSTARTKEY~
~~RECSTOPKEY~
~~REC_MAXCHAR~
~~REC_MAX_ON~
~~REQFILE~
~~REQPATH~
~~REQUESTKEY~
~~STOPKEY~
~~System~requirements~
~~Usage~
~~WAIT~
~File
~Hotkey~
~Info-Requester~
~InputEvents~
~\d~Inputevent
Arexx
CLI_ASCSEND
DELAY
DELAY
DELAYCHAR
DELAYCHAR
MUI
MUI-ASCII_SEND
RECFILE

RECSTOPKEY
REC_MAXCHAR
REC_MAX_ON
REQUESTKEY
REQFILE
REQPATH