

# **Connectline**

Ralf Range

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> Connectline	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY	Ralf Ramage	February 12, 2023
<i>SIGNATURE</i>		

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Connectline</b>	<b>1</b>
1.1	Dokumentation der cl_rexx.library . . . . .	1
1.2	Allgemeine Funktionen . . . . .	2
1.3	Befehle für Zugriffe auf Brettdaten . . . . .	6
1.4	Befehle für Zugriff auf Userdaten . . . . .	20
1.5	Befehle für Zugriffe auf Systeme . . . . .	33
1.6	Funktionen für Onlinebefehle . . . . .	41
1.7	Befehle für Zugriff auf Nachrichten . . . . .	47
1.8	Befehle für Zugriffe auf die Ports . . . . .	50
1.9	Befehle zum Auslesen von Defaultpfaden . . . . .	51
1.10	Index . . . . .	52
1.11	ARexx-Kommandos des CL-SERVER . . . . .	60

---

# Chapter 1

## Connectline

### 1.1 Dokumentation der cl\_rexx.library

Die cl\_rexx.library

Einführung:

-----

Die Connectline cl\_rexx.library stellt ein mächtiges Werkzeug dar, mit welchem Sie leistungsfähige AREXX-Skripte für die Wartung Ihres Systems und die Onlineunterstützung der User erstellen können.

Diese Dokumentation erläutert sämtliche Funktionen der cl\_rexx.library in thematisch zusammengehörigen Kapiteln. Folgende Bereiche werden von der cl\_rexx.library unterstützt:

1. Allgemeine~Funktionen
  2. Befehle~für~Zugriffe~auf~Brettdaten
  3. Befehle~für~Zugriffe~auf~Userdaten
  4. Befehle~für~Zugriffe~auf~Systeme
  5. Funktionen~für~Onlinebefehle
  6. Befehle~für~Zugriffe~auf~Nachrichten
  7. Befehle~für~Zugriffe~auf~die~Ports
  8. Befehle~zum~Auslesen~von~Defaultpfaden
- Funktionstypen der cl\_rexx.library

-----

Die Funktionen der cl\_rexx.library sind verschiedenen Funktionstypen zugeordnet und ihre Zugehörigkeit ist an folgenden Prefixen erkennbar:

CLGET...  
- Liefert Wert von Connectline

CLSET...  
- Setzt Wert bei Connectline

CLIS...~  
- Abfragen von Flags, Wert immer "1" (wahr) oder "0" (falsch).

CLONL...  
- Macht Ein-/Ausgaben in der Shell. Je nach Funktion wird ein Wert zurückgeliefert.

## 1.2 Allgemeine Funktionen

### 1. Allgemeine Funktionen:

- o CLInfo()
  - o CLGET\_OwnSystemName()
- o CLGET\_OwnSystemAddress()
- o CLGET\_GroupName(<gruppennr>)
- o CLGET\_TransferList(<variable>)
- o CLGET\_PackerList(<variable>)
- o CL\_SendMailFull(<absender>, <empfänger>, <sourcefile>, <commentfile>, <shortcomment>. <subject>, <contenttype>, <sortinpri>)
- o CL\_LogAdd(<priorität>, <ID>, <text>)
- o CL\_LogDelete()
- o CL\_LogRename(<dateiname>)
- o CLGET\_FileRequest([pfad])
- o CLGET\_FrontScreenName()
- o CL\_GetText

- (<textdatei>,<varname>)
  - o
    - CL\_PutStr
    - (<text>)
  - o CLInfo()
 

Testfunktion der cl\_rexx.library. Sie liefert die aktuelle Versionsnummer und einen Copyrightvermerk zurück.
  - o CLGET\_OwnSystemName()
 

Diese Funktion liefert den eigenen Systemnamen ohne die Domainangabe zurück.

Beispiel:

```
systemname=CLGET_OwnSystemName()
say systemname
```
  - o CLGET\_OwnSystemAddress()
 

Diese Funktion dient dem Ermitteln des eigenen Systemnamens inkl. der Domainangabe. Es wird immer die Hauptdomain zurückgegeben, auch wenn das System an mehreren Domains teilnimmt.

Beispiel:

```
/* Ermitteln der Systemadresse und Zerlegen in System- und
   Domainname */

adresse=CLGET_OwnSystemAddress()
system=left(adresse,pos('.',adresse)-1)
domain=right(adresse,length(adresse)-pos('.',adresse))
say 'Systemname:' system
say 'Domainname:' domain
```
  - o CLGET\_GroupName(<gruppennr>)
 

Mit dieser Funktion erhält man den Gruppennamen von <gruppennr> zurück. Die Übergabe eines Parameters > 0 ist zwingend erforderlich. Ist für <gruppennr> kein Gruppename spezifiziert, wird ein Leerstring zurückgegeben.

Beispiel:

```
say CLGET_GroupName(1)
```
  - o CLGET\_TransferList(<variable>)
 

Diese Funktion liefert die Anzahl der eingestellten Transferprotokolle
-

zurück. Bei dem Parameter <variable> handelt es sich um den Namen einer STEM-Variablen, welche nach erfolgreichem Aufruf der Funktion die Namen der Protokolle enthält. Erster Index ist 0, letzter Index ist <Rückgabewert>-1.

o CLGET\_PackerList (<variable>)

Diese Funktion liefert die Anzahl der eingestellten Packer zurück. Bei dem Parameter <variable> handelt es sich um den Namen einer STEM-Variablen, welche nach erfolgreichem Aufruf der Funktion die Namen der Packer enthält. Erster Index ist 0, letzter Index ist <Rückgabewert>-1.

o CL\_SendMailFull (<absender>, <empfänger>, <sourcefile>, <commentfile>, <shortcomment>. <subject>, <contenttype>, <sortinpri>)

Diese Funktion dient dem externen Verschicken von Nachrichten. Siehe auch CLONL\_SendMail().

Parameter:

absender - Username des Absenders. Es wird nicht geprüft, ob der User im System existiert. System- und Domainangabe werden automatisch ersetzt, der Realname bleibt erhalten.

empfänger - Empfänger der Nachricht.

sourcefile - Pfad und Name des zu verschickenden Textes/Files.

commentfile- Parameter mit Pfad und Name des langen Kommentars (KOM:-Header). Soll kein langer Kommentar in die Mail eingefügt werden, ist ein Leerstring zu übergeben.

shortcomment- Angabe des kurzen Kommentars (ZUSAMMENFASSUNG:-Header). Wird als String übergeben, ist dieser Header nicht gewünscht, erfolgt die Übergabe eines Leerstrings.

subject - String mit dem Betreff der Nachricht.

contenttype- Soll die Nachricht als Binärfile verschickt werden, geben sie hier 'BIN' an, ansonsten 'TEXT'.

sortinpri - Priorität der Mail. Sie beeinflusst die Reihenfolge, in welcher der Sorter die vorliegenden Puffer bearbeitet. Höchste Priorität (Eilmal) ist 'A'.

Beispiel:

```
cl_sendmailfull(test,root,'s:startup-sequence',,,,'subject',TEXT,'A')
```

```
/* Verschickt das File s:startup-sequence als Textnachricht an den User 'root'. Als Absender wird der User 'test' angegeben. Der Be-
```

treff der Nachricht lautet 'subject', die Nachricht hat höchste Priorität. Langer und kurzer Kommentar werden nicht gesetzt. \*/

o CL\_LogAdd(<priorität>,<ID>,<text>)

Fügt einen Eintrag ins Logfile von Connectline 5.0 ein.

Parameter:

priorität - Priorität des Logeintrags (0<=x<=10)  
ID - Kennung des Prozesses  
text - Einzutragender Text

Beispiel:

```
call CL_LogAdd(5,'CL_REXX','Das ist ein Test.')
```

o CL\_LogDelete()

Löscht das Logfile.

o CL\_LogRename(<dateiname>)

Benennt das Logfile in <dateiname> um. Ein eventueller Pfad muß angegeben werden.

o CLGET\_FileRequest([<pfad>])

Diese Funktion öffnet einen Filerequester und liefert den Namen und Pfad des angewählten Eintrages zurück. Optional kann mit <pfad> ein Defaultpfad angegeben werden. Multiselect ist nicht möglich.

o CLGET\_FrontScreenName()

Diese Funktion liefert den Namen des vordersten Public Screens zurück.

o CL\_GetText(<textdatei>,<varname>)

Es wird ein File zeilenweise in eine STEM-Variable eingelesen. Die Anzahl der eingelesenen Zeilen wird zurückgegeben.

Parameter:

textdatei - Pfad und Name des einzulesenden Textes  
varname - Name der STEM-Variable ohne Index

Beispiel:

```
anzahl=CL_GetText('s:startup-sequence',zeile)
do x=0 to anzahl-1
  say zeile.x
```

end

- o CL\_PutStr(<text>)

Gibt einen Text direkt auf dem Bildschirm aus, es wird kein LF ausgeführt.

## 1.3 Befehle für Zugriffe auf Brettdaten

### 2. Befehle für Zugriffe auf Brettdaten

---

Allgemeine Anmerkungen:

---

Wenn versucht wird, auf ein nicht existentes Brett zuzugreifen, wird beim Lesen generell ein Leerstring "" zurückgegeben, und beim Schreiben False "0".

Bretter werden generell so angegeben, wie sie im Bretteditor auftauchen. Ein führendes Slash ("/") darf nicht übergeben werden.

#### 2.1

Allgemeine~Befehle

2.2

Befehle~zum~Auslesen~der~Brettdaten

2.3

Befehle~zum~Setzen~der~Brettdaten

2.4

Befehle~zum~Auslesen~und~Verändern~der~Verteilerliste

2.5

Befehle~zum~Testen~der~Zugriffsberechtigungen

2.6

Befehle~zum~Abfragen~von~Brettflags

2.7

Befehle~zum~Setzen~von~Brettflags

2.1 Allgemeine Befehle

---

- o

CL\_SaveBoardlist

()

- o

CL\_BoardNew

(<brettname>)

- o

CL\_BoardDelete

(<brettname>)

- o CL\_SaveBoardlist ()

---

Speichert die Brettliste.

o `CL_BoardNew(<brettname>)`

Diese Funktion legt ein neues Brett an. Initialisieren Sie dieses Brett anschließend mit allen `CLSET_BoardX()`-Funktionen (siehe Abschnitt 2.3 und 2.7 in diesem Kapitel), da diese Funktion lediglich Defaultwerte vorgibt.

Die Defaultwerte sind:

- Als Pfad wird der Defaultpfad für Bretter eingesetzt, mit der Ergänzung auf diesen Brettnamen.
- Lese- und Schreibzugriff werden auf Level 99 begrenzt.
- Als Brettkommentar wird "Neues Brett (cl\_rexx)" eingetragen.
- Das Flag 'Realname' wird auf TRUE gesetzt.

Alle anderen Brettparameter sind entweder 0 oder deaktiviert.

Beispiel:

```
if (CL_BoardNew('INTERN/TEST') then
  say 'Das Brett /INTERN/TEST wurde angelegt.'
else
  say 'Es ist ein Fehler aufgetreten!'
```

o `CL_BoardDelete(<brettname>)`

Das Brett <brettname> wird aus der Brettliste gelöscht.

Beispiel:

```
if (CL_BoardDelete('INTERN/TEST') then
  say 'Das Brett /INTERN/TEST wurde gelöscht.'
else
  say 'Das Brett /INTERN/TEST existiert nicht!'
```

## 2.2 Befehle zum Auslesen der Brettdateien

---

- o `CLGET_BoardList`  
(<varname> [, <pattern>])
- o `CLIS_BoardExists`

- o (<brettname>)
  - o CLGET\_BoardNumberOf  
( )
  - o CLGET\_BoardNext  
(<brettname>)
  - o CLGET\_BoardPrevious  
(<brettname>)
  - o CLGET\_BoardPrefix  
(<prefix>)
  - o CLGET\_BoardNewsgroupName  
(<brettname>)
  - o CLGET\_BoardFTSName  
(<brettname>)
  - o CLGET\_BoardPath  
(<brettname>)
  - o CLGET\_BoardAdmin  
(<brettname>)
  - o CLGET\_BoardPri  
(<brettname>)
  - o CLGET\_BoardWritePri  
(<brettname>)
  - o CLGET\_BoardMaxNet  
(<brettname>)
  - o CLGET\_BoardAbout  
(<brettname>)
  - o CLGET\_BoardCensor  
(<brettname>)
  - o CLGET\_BoardHoldKB  
(<brettname>, <maxkb>)
  - o CLGET\_BoardHoldDays  
(<brettname>, <maxdays>)
  - o CLGET\_BoardLastWrite  
(<brettname>)
  - o CLIS\_BoardGroup  
(<brettname>, <group>)
  
  - o CLGET\_BoardList (<varname> [, <pattern>])
-

Legt eine vollständige Brettliste in der STEM-Variablen <varname> an. Der größte Index wird zurückgegeben, der niedrigste Index ist 0. Wird ein Muster <pattern> angegeben, so werden nur diese Bretter, die zu dem Muster passen, berücksichtigt (z.B. 'CLC/#?').

Parameter:

varname - Bezeichnung der STEM-Variable ohne Index  
 pattern - Optionaler Parameter mit Suchmuster

Beispiel:

```
anzahl1=CLGET_BoardList(alleBretter)
anzahl2=CLGET_BoardList(alleCLCBretter)
do x = 1 to anzahl1-1
  say 'Brett Nummer 'x': 'alleBretter.x
end
do x= 1 to anzahl2-1
  say 'CLC-Brett Nummer 'x': 'alleCLCBretter.x
end
```

#### o CLIS\_BoardExists(<brettname>)

Diese Funktion testet, ob das Brett <brettname> in der Brettliste existiert.

Beispiel:

```
if (CLIS_BoardExists('INTERN/TEST') then
  say '/INTERN/TEST ist ein vorhandenes Brett.'
else
  say '/INTERN/TEST ist ein unbekanntes Brett.'
```

#### o CLGET\_BoardNumberOf()

Liefert die Anzahl der eingetragenen Bretter zurück.

#### o CLGET\_BoardNext(<brettname>)

Diese Funktion gibt den Namen des Brettes zurück, welches <brettname> folgt. Möchte man das erste Brett der gesamten Brettliste ermitteln, übergibt man als Parameter einen Leerstring. Ist das letzte Brett der Liste erreicht, wird von der Funktion ein Leerstring zurückgegeben.

Beispiel:

```
/* Einlesen der gesamten Brettliste in eine STEM-Variable */

x=
  CLGET_BoardNumberOf
  ()
brett.1=CLGET_BoardNext()
do i=2 to x
```

```
brett.i=CLGET_BoardNext(brett.(i-1))
end
```

o CLGET\_BoardPrevious(<brettname>)

Diese Funktion gibt den Namen des Brettes zurück, welches <brettname> vorhergeht. Möchte man das letzte Brett der gesamten Brettliste ermitteln, übergibt man als Parameter einen Leerstring. Ist das erste Brett der Liste erreicht, wird von der Funktion ein Leerstring zurückgegeben.

Beispiel:

```
/* Einlesen der gesamten Brettliste in eine STEM-Variable */

x=
    CLGET_BoardNumberOf
    ()
brett.x=CLGET_BoardPrevious()
do i=(x-1) to 1 by -1
    brett.i=CLGET_BoardPrevious(brett.(i+1))
end
```

o CLGET\_BoardPrefix(<prefix>)

Diese Funktion versucht, <prefix> zu einem existierenden Brettnamen zu ergänzen.

Beispiel:

```
say 'CLGET_BoardPrefix('CLC/')
/* Als Ergebnis wird das erste Brett der Hierarchie /CLC zurück-
gegeben. */
```

o CLGET\_BoardNewsgroupName(<brettname>)

Liefert den Newsgroupnamen dieses Brettes (rfc-Modul).

o CLGET\_BoardFTSName(<brettname>)

Liefert den FTS-Namen dieses Brettes (FIDO-Modul).

o CLGET\_BoardPath(<brettname>)

Diese Funktion gibt den Dos-Pfad eines Brettes zurück.

o CLGET\_BoardAdmin(<brettname>)

Liefert den Usernamen des Brettverwalters des angegebenen Brettes.

---

o CLGET\_BoardPri (<brettname>)

Liefert die Mindestpriorität zum Lesen des Brettes.

o CLGET\_BoardWritePri (<brettname>)

Liefert die Mindestpriorität zum Beschreiben des Brettes.

o CLGET\_BoardMaxNet (<brettname>)

Gibt die maximale zugelassene Nachrichtengröße in diesem Brett zurück.

o CLGET\_BoardAbout (<brettname>)

Liefert den Kommentar dieses Brettes.

o CLGET\_BoardCensor (<brettname>)

Liefert den Usernamen des Zensors dieses Brettes.

o CLGET\_BoardHoldKB (<brettname>, <maxkb>)

Liefert das maximale Haltevolumen in KB für den Cruncher in diesem Brett.

o CLGET\_BoardHoldDays (<brettname>, <maxdays>)

Liefert die maximale Haltezeit in Tagen für den Cruncher in diesem Brett.

o CLGET\_BoardLastWrite (<brettname>)

Liefert den Zeitpunkt zurück, an welchem der letzte Schreibzugriff in diesem Brett stattgefunden hat. Es wird ein String mit zwei Worten zurückgegeben; das erste Wort enthält das Datum im Format DD.MM.YYYY, das zweite Wort die Uhrzeit im Format HH:MM:SS.

Beispiel:

```
/* Ermitteln und zerlegen des letzten Schreibzugriffes im Brett
   /CLC/USER/FRAGEN. */
```

```
all=clget_BoardLastWrite('CLC/USER/FRAGEN')
datum=translate(word(all,1),' ','.')
zeit=translate(word(all,2),' ','.')
Tag=word(datum,1); Monat=word(datum,2); Jahr=word(datum,3)
Stunde=word(zeit,1); Minute=word(zeit,2); Sekunde=word(zeit,3)
```

- o CLIS\_BoardGroup(<brettname>,<group>)

Diese Funktion dient der Abfrage, ob ein Brett einer bestimmten Gruppe angehört.

Parameter:

brettname - Name des Brettes  
group - Nummer der Gruppe

Beispiel:

```
/* Heiteres Gruppenraten :-) */

brett='CLC/USER/FRAGEN'
options prompt 'Ihr Tip: '
do forever
  pull gruppennr
  select
    when gruppennr=-1 then leave
    when ~CLIS_Boardgroup(brett,gruppennr) then do
      say 'Nein, /CLC/USER/FRAGEN ist nicht in'
      say 'der Gruppe "'
      CLGet_GroupName
      (gruppennr)'"'
    end
    otherwise do
      say 'Sie haben richtig geraten!'
      exit
    end
  end
end
```

### 2.3 Befehle zum Setzen der Brettdaten

---

Bei Erfolg wird generell '1' zurückgeliefert, bei Mißerfolg '0'.

Beispiel für die Benutzung dieser Funktionen:

```
if (funktionsaufruf) then
  ... /* Aufruf war erfolgreich */
else
  ... /* Aufruf war nicht erfolgreich */
```

- o CLSET\_BoardNewsgroupName  
(<brettname>,<ngname>)
- o

- o CLSET\_BoardFTSName  
(**<brettname>**,**<ftsname>**)
- o CLSET\_BoardAdmin  
(**<brettname>**,**<admin>**)
- o CLSET\_BoardPri  
(**<brettname>**,**<pri>**)
- o CLSET\_BoardWritePri  
(**<brettname>**,**<pri>**)
- o CLSET\_BoardMaxNet  
(**<brettname>**,**<maxnet>**)
- o CLSET\_BoardAbout  
(**<brettname>**,**<infotext>**)
- o CLSET\_BoardCensor  
(**<brettname>**,**<censor>**)
- o CLSET\_BoardHoldKB  
(**<brettname>**)
- o CLSET\_BoardHoldDays  
(**<brettname>**)
- o CLSET\_BoardGroup  
(**<brettname>**,**<gruppennr>** [**,<flag>**])

- o CLSET\_BoardNewsgroupName(**<brettname>**,**<ngrname>**)

Setzt den Newsgroupnamen dieses Brettes (für rfc-Modul).

Parameter:

**brettname** - Name des Brettes  
**ngrname** - gewünschter Newsgroupname

- o CLSET\_BoardFTSName(**<brettname>**,**<ftsname>**)

Setzt den FTS-Namen dieses Brettes (für FIDO-Modul).

Parameter:

**brettname** - Name des Brettes  
**ftsname** - gewünschter Echoname

- o CLSET\_BoardAdmin(**<brettname>**,**<admin>**)

Setzt den Brettverwalter des Brettes.

---

Parameter:

brettname - Name des Brettes  
admin - Username des Brettverwalters

o CLSET\_BoardPri(<brettname>,<pri>)

Setzt die erforderliche Mindestpriorität zum Lesen des Brettes.

Parameter:

brettname - Name des Brettes  
pri - Priorität (0<=pri<=99)

o CLSET\_BoardWritePri(<brettname>,<pri>)

Setzt die erforderliche Mindestpriorität zum Beschreiben des Brettes.

Parameter:

brettname - Name des Brettes  
pri - Priorität (0<=pri<=99)

o CLSET\_BoardMaxNet(<brettname>,<maxnet>)

Setzt die maximal zugelassene Nachrichtengröße des Brettes.

Parameter:

brettname - Name des Brettes  
maxnet - Maximallänge in Byte oder 0.

o CLSET\_BoardAbout(<brettname>,<infotext>)

Setzt den Kommentar dieses Brettes.

Parameter:

brettname - Name des Brettes  
infotext - Brettkommentar

o CLSET\_BoardCensor(<brettname>,<censor>)

Setzt den Zensor dieses Brettes.

Parameter:

brettname - Name des Brettes  
censor - Username des Zensors

o CLSET\_BoardHoldKB(<brettname>)

---

Setzt das maximale Brettvolumen in Kilobyte für den Cruncher.

- o CLSET\_BoardHoldDays(<brettname>)

Setzt die maximale Haltezeit in Tagen für den Cruncher.

- o CLSET\_BoardGroup(<brettname>,<gruppennr> [,<flag>])

Trägt ein Brett in eine Gruppe ein oder trägt es aus einer Gruppe aus.

Parameter:

brettname - Name des Brettes  
 gruppennr - Nummer der Gruppe  
 flag - Optionales Flag; ist dies angegeben und hat den Wert 0, wird das Brett aus der Gruppe ausgetragen. Fehlt dieses Flag beim Aufruf, wird ein Eintragen in die Gruppe versucht.

Beispiel:

```
if (CLSET_BoardGroup('CLC/USER/FRAGEN',1) then do
  say '/CLC/USER/FRAGEN in Gruppe 1 eingetragen!'

  call CLSET_BoardGroup('CLC/USER/FRAGEN',1,'0')
  say '/CLC/USER/FRAGEN aus Gruppe 1 ausgetragen!'
end
```

## 2.4 Befehle zum Auslesen und Verändern der Verteilerliste

---

- o CLGET\_BoardBoxList  
(<brettname>, <varname>)

- o CLGET\_BoardNetCount  
(<brettname>)

- o CLSET\_BoardAddBox  
(<brettname>,<system>)

- o CLSET\_BoardDelBox  
(<brettname>,<system>)

- o CLGET\_BoardBoxList(<brettname>, <varname>)

Liefert die Anzahl der Systeme zurück, die das Brett <brettname> be-

---

stellt haben. Die Namen der Systeme werden in der STEM-Variablen <varname> abgelegt. Das erste System ist in der Variablen <varname>.0 zu finden.

Beispiel:

```
options prompt "Gewünschtes Brett:"
pull brett
anzahl=CLGET_BoardBoxList(brett,system)
say anzahl' Systeme im Verteiler von 'brett':'
do x=0 to anzahl-1
  say ' 'system.x
end
```

o CLGET\_BoardNetCount(<brettname>)

Liefert die Anzahl der im Verteiler des Brettes <brettname> eingetragenen Systeme.

Hinweis: Wenn Sie nicht nur an der Zahl der Systeme im Verteiler interessiert sind, sondern den Verteilerstring mittels

```
CLGET_BoardBoxList
() sowieso auslesen, erhalten sie die Anzahl über
die interne ARexx-Funktion words() auf einem schnelleren Weg.
```

o CLSET\_BoardAddBox(<brettname>,<system>)

Fügt ein neues System in die Verteilerliste des Brettes ein. Es wird nicht überprüft, ob das System in der Systemliste eingetragen ist.

Parameter:

```
brettname - Name des Brettes
system    - Name des Systems/des Points.
```

o CLSET\_BoardDelBox(<brettname>,<system>)

Löscht ein System aus der Verteilerliste des Brettes. Es wird nicht überprüft, ob das System in der Systemliste eingetragen ist.

Parameter:

```
brettname - Name des Brettes
system    - Name des Systems/des Points.
```

## 2.5 Befehle zum Testen der Zugriffsberechtigungen

---

o

- CLGET\_BoardValidateUser  
(<brettname>,<user>)
  - o
  - CLGET\_BoardValidateBox  
(<brettname>,<system>)

- o CLGET\_BoardValidateUser(<brettname>,<user>)

Überprüft, ob und wie der User Zugriff auf das Brett hat. Diese Funktion gibt einen der folgenden Strings zurück:

NONE	User hat keinerlei Zugriff.
ACCESS	User hat Zugriff.
CENSORED	User hat Zugriff und wird zensiert.
ISADMIN	User hat Zugriff und ist Brettverwalter.
ISCENSOR	User hat Zugriff und ist Zensor.

- o CLGET\_BoardValidateBox(<brettname>,<system>)

Überprüft, ob das System Zugriff auf das Brett hat.

Beispiel:

```
if (CLGET_BoardValidateBox(brett,system)
    say upper(system)' hat Zugriff auf 'brett'!'
else
    say upper(system)' hat *keinen* Zugriff auf 'brett'!'
```

## 2.6 Befehle zum Abfragen von Brettflags

---

Ist das jeweilige Flag gesetzt, liefern die Funktionen '1' zurück, ist es nicht gesetzt, eine '0'.

- o
  - CLIS\_BoardBin  
(<brettname>)
  - o
  - CLIS\_BoardText  
(<brettname>)
  - o
  - CLIS\_BoardMapsAllowed  
(<brettname>)
  - o
  - CLIS\_BoardAutoEntry  
(<brettname>)
  - o
  - CLIS\_BoardRealname  
(<brettname>)

- o CLIS\_BoardDLRatio (<brettname>)
  - o CLIS\_BoardAutoRemove (<brettname>)
  - o CLIS\_BoardExternal (<brettname>)
  - o CLIS\_BoardRouteOnly (<brettname>)
  - o CLIS\_BoardNoScan (<brettname>)
  
  - o CLIS\_BoardBin (<brettname>)  
Abfrage, ob Binärnachrichten im Brett erlaubt sind.
  
  - o CLIS\_BoardText (<brettname>)  
Abfrage, ob Textnachrichten im Brett erlaubt sind.
  
  - o CLIS\_BoardMapsAllowed (<brettname>)  
Abfrage, ob per Maps auf das Brett zugegriffen werden kann.
  
  - o CLIS\_BoardAutoEntry (<brettname>)  
Abfrage, ob das Brett automatisch eingetragen wurde.
  
  - o CLIS\_BoardRealname (<brettname>)  
Abfrage, ob im Brett ein Realnamenzwang besteht.
  
  - o CLIS\_BoardDLRatio (<brettname>)  
Abfrage, ob im Brett eine Up-/Downloadratio aktiv ist.
  
  - o CLIS\_BoardAutoRemove (<brettname>)  
Abfrage, ob das Brett automatisch ausgetragen wird.
  
  - o CLIS\_BoardExternal (<brettname>)  
Abfrage, ob es sich um ein extern ausgelagertes Brett handelt.
-

- o CLIS\_BoardRouteOnly(<brettname>)

Abfrage, ob das Brett nur durchgeroutet wird.

- o CLIS\_BoardNoScan(<brettname>)

Abfrage, ob bei dem Brett das Datum konstant ist (z.B. CD-ROM's).

## 2.7 Befehle zum Setzen von Brettflags

---

Bei den Funktionen dieses Abschnittes sind die zu übergebenden Parameter von Funktion zu Funktion identisch und haben folgenden Inhalt:

Parameter:

brettname - Name des Brettes  
flag - '0' bedeutet, das Flag wird gelöscht (FALSE)  
          '1' bedeutet, das Flag wird gesetzt (TRUE)

- o CLSET\_BoardBin  
(<brettname>,<flag>)
  - o CLSET\_BoardText  
(<brettname>,<flag>)
  - o CLSET\_BoardAutoEntry  
(<brettname>,<flag>)
  - o CLSET\_BoardRealname  
(<brettname>,<flag>)
  - o CLSET\_BoardDLRatio  
(<brettname>,<flag>)
  - o CLSET\_BoardAutoRemove  
(<brettname>,<flag>)
  - o CLSET\_BoardExternal  
(<brettname>,<flag>)
  - o CLSET\_BoardRouteOnly  
(<brettname>,<flag>)
  - o CLSET\_BoardNoScan  
(<brettname>,<flag>)
-

- o CLSET\_BoardBin(<brettname>,<flag>)  
Flag, ob Binärnachrichten im Brett erlaubt sind.
- o CLSET\_BoardText(<brettname>,<flag>)  
Flag, ob Textnachrichten im Brett erlaubt sind.
- o CLSET\_BoardAutoEntry(<brettname>,<flag>)  
Flag, ob das Brett wie ein automatisch eingetragenes Brett behandelt werden soll.
- o CLSET\_BoardRealname(<brettname>,<flag>)  
Flag, ob im Brett ein Realnamenzwang existieren soll.
- o CLSET\_BoardDLRatio(<brettname>,<flag>)  
Flag, ob im Brett eine Up-/Downloadratio gelten soll.
- o CLSET\_BoardAutoRemove(<brettname>,<flag>)  
Flag, ob das Brett automatisch ausgetragen werden soll.
- o CLSET\_BoardExternal(<brettname>,<flag>)  
Setzt je nach Flag, ob das Brett extern ausgelagert ist.
- o CLSET\_BoardRouteOnly(<brettname>,<flag>)  
Flag, ob das Brett nur geroutet wird.
- o CLSET\_BoardNoScan(<brettname>,<flag>)  
Flag, ob bei dem Brett ein konstantes Datum herrschen soll.

## 1.4 Befehle für Zugriff auf Userdaten

### 3. Befehle für Zugriff auf Userdaten

---

Allgemeine Anmerkungen:

---

Wenn versucht wird, auf einen nicht existenten User zuzugreifen, wird beim

---

Lesen generell ein Leerstring zurückgegeben, beim Schreiben FALSE ('0').

Es ist zwar möglich, auf User zuzugreifen, die sich gerade online im System aufhalten, aber es wird davon abgeraten. Dies liegt darin begründet, daß beim Logoff des Users seine Userdaten wieder aus dem RAM gespeichert werden und Änderungen mittels cl\_rexx hierdurch hinfällig werden.

### 3.1

Allgemeine~Befehle

3.2

Befehle~zum~Auslesen~der~Userdaten

3.3

Befehle~zum~Setzen~der~Userdaten

3.4

Befehle~zum~Lesen~der~Userflags

3.1 Allgemeine Befehle

- 
- o CLIS\_UserExists  
(<username>)
  - o CLIS\_UserPoint  
(<username>)
  - o CL\_UserNew  
(<username>, <passwort>)
  - o CLGET\_UserNumberOf  
( )
  - o CLGET\_Username  
(<index>)

#### o CLIS\_UserExists (<username>)

Diese Funktion liefert als Rückgabewert TRUE, wenn <username> eingetragener User ist, FALSE, wenn dies nicht der Fall ist.

#### o CLIS\_UserPoint (<username>)

Liefert TRUE wenn der User einen Point in Ihrem System hat, ist dies nicht der Fall, wird FALSE zurückgegeben.

#### o CL\_UserNew (<username>, <passwort>)

Es wird ein neuer User unter <username> mit dem Passwort <passwort> eingetragen. Bei Erfolg wird '0' zurückgegeben, ansonsten '1'. Danach sollten mit den CLSET\_UserX()-Befehlen (siehe Abschnitt

3.3

)

---

alle weiteren Daten ergänzt werden.

- o CLGET\_UserNumberOf()

Diese Funktion gibt die Anzahl aller eingetragenen User zurück.

- o CLGET\_Username(<index>)

Diese Funktion liefert den Usernamen des Users <index> zurück.

Beispiel:

```
/* Erstellen einer Liste aller User */  
  
anz=  
    CLGET_UserNumberOf  
    (  
do x=1 to anz  
    user.x=CLGET_Username(x)  
end
```

### 3.2 Befehle zum Auslesen der Userdaten

---

- o CLGET\_UserPriority  
(<username>)
  - o CLGET\_UserPath  
(<username>)
  - o CLIS\_UserGroup  
(<username>,<gruppennr>)
  - o CLGET\_UserRealname  
(<username>)
  - o CLGET\_UserAddress  
(<username>)
  - o CLGET\_UserTelephone  
(<username>)
  - o CLGET\_UserSysopInfo  
(<username>)
  - o CLGET\_UserMessage  
(<username>)
  - o CLGET\_UserCopyTo  
(<username>)
  - o CLGET\_UserTerminal
-

- o (<username>)
- o CLGET\_UserLanguage  
(<username>)
- o CLGET\_UserKeyboard  
(<username>)
- o CLGET\_UserNumberOfCalls  
(<username>)
- o CLGET\_UserTimeout  
(<username>)
- o CLGET\_UserCredit  
(<username>)
- o CLGET\_UserLines  
(<username>)
- o CLGET\_UserProtocol  
(<username>)
- o CLGET\_UserMaxKB  
(<username>)
- o CLGET\_UserMaxDays  
(<username>)
- o CLGET\_UserLastCall  
(<username>)
- o CLGET\_UserUploads  
(<username>)
- o CLGET\_UserDownloads  
(<username>)
- o CLGET\_UserFreeFiles  
(<username>)
- o CLGET\_UserFreeKByte  
(<username>)
- o CLGET\_UserRate  
(<username>)
- o CLGET\_UserPointname  
(<username>)
  
- o CLGET\_UserPriority(<username>)

Liefert die Priorität des Users zurück.

---

- o CLGET\_UserPath(<username>  
Liefert den Pfad des Users zurück.
  - o CLIS\_UserGroup(<username>,<gruppennr>  
Liefert '1', wenn der User in der Gruppe <gruppennr> ist, sonst '0'.
  - o CLGET\_UserRealname(<username>  
Liefert den Realnamen des Users zurück.
  - o CLGET\_UserAddress(<username>  
Liefert die Adresse des Users zurück.
  - o CLGET\_UserTelephone(<username>  
Liefert die Telefonnummer des Users zurück.
  - o CLGET\_UserSysopInfo(<username>  
Liefert die Sysopinfo des Users zurück.
  - o CLGET\_UserMessage(<username>  
Liefert Userinfo des Users zurück.
  - o CLGET\_UserCopyTo(<username>  
Liefert den Vertreter des Users zurück.
  - o CLGET\_UserTerminal(<username>  
Liefert entweder "ANSI", "VT100" oder "TTY" zurück, je nach Terminaleinstellung des Users.
  - o CLGET\_UserLanguage(<username>  
Liefert Spracheinstellung des Users zurück.
  - o CLGET\_UserKeyboard(<username>  
Liefert den eingestellten Zeichensatz zurück ("IBM" oder "ISO").
-

- o `CLGET_UserNumberOfCalls(<username>)`

Gibt die Gesamtzahl aller Anrufe des Users zurück.
  - o `CLGET_UserTimeout(<username>)`

Liefert den Zeitlimit des Users zurück (Minuten).
  - o `CLGET_UserCredit(<username>)`

Liefert den erlaubten Kredit für den User zurück.
  - o `CLGET_UserLines(<username>)`

Liefert die eingestellte Zeilenzahl des Users zurück.
  - o `CLGET_UserProtocol(<username>)`

Liefert das eingestellte Übertragungsprotokoll zurück.
  - o `CLGET_UserMaxKB(<username>)`

Liefert das maximale Haltevolumen in KB des Postfaches des Users für den Cruncher zurück.
  - o `CLGET_UserMaxDays(<username>)`

Liefert die maximale Haltezeit in Tagen des Postfaches des Users für den Cruncher zurück.
  - o `CLGET_UserLastCall(<username>)`

Liefert den Zeitpunkt des letzten Anrufes des Users zurück. Es wird ein String mit 2 Worten zurückgegeben; das erste Wort beinhaltet das Datum im Format DD.MM.YYYY, das zweite die Uhrzeit im Format HH:MM:SS.
  - o `CLGET_UserUploads(<username>)`

Liefert die Uploads des Users in Kilobyte zurück.
  - o `CLGET_UserDownloads(<username>)`

Liefert Downloads des Users in Kilobyte zurück.
  - o `CLGET_UserFreeFiles(<username>)`

Liefert die freien Files des Users zurück.
-

- o CLGET\_UserFreeKByte(<username>  
Liefert den Downloadfreiraum des Users zurück.
- o CLGET\_UserRate(<username>  
Liefert die Up-/Downloadratio des Users zurück.
- o CLGET\_UserPointname(<username>  
Liefert den Pointnamen des Users zurück.

### 3.3 Befehle zum Setzen der Userdaten

---

Bei Erfolg wird generell '1' zurückgeliefert, bei Mißerfolg '0'.

- o CLSET\_UserPassword  
(<username>, <passwort>)
  - o CLSET\_UserPriority  
(<username>, <priorität>)
  - o CLSET\_UserPath  
(<username>, <pfad>)
  - o CLSET\_UserGroup  
(<username>, <gruppennr> [, <flag>])
  - o CLSET\_UserRealname  
(<username>, <realname>)
  - o CLSET\_UserAddress  
(<username>, <adresse>)
  - o CLSET\_UserTelephone  
(<username>, <telefon>)
  - o CLSET\_UserSysopInfo  
(<username>, <sysopinfo>)
  - o CLSET\_UserMessage  
(<username>, <message>)
  - o CLSET\_UserCopyTo  
(<username>, <vertreter>)
  - o CLSET\_UserTerminal  
(<username>, <terminal>)
  - o
-

- o CLSET\_UserLanguage  
(<username>, <sprache>)
  - o CLSET\_UserKeyboard  
(<username>, <zeichensatz>)
  - o CLSET\_UserTimeout  
(<username>, <timeout>)
  - o CLSET\_UserAccount  
(<username>, <konto>)
  - o CLSET\_UserCredit  
(<username>, <kredit>)
  - o CLSET\_UserLines  
(<username>, <zeilenzahl>)
  - o CLSET\_UserProtocol  
(<username>, <protokoll>)
  - o CLSET\_UserMaxKB  
(<username>, <maxkb>)
  - o CLSET\_UserMaxDays  
(<username>, <maxdays>)
  - o CLSET\_UserUploads  
(<username>, <uploads>)
  - o CLSET\_UserDownloads  
(<username>, <downloads>)
  - o CLSET\_UserFreeFiles  
(<username>, <freefiles>)
  - o CLSET\_UserFreeKByte  
(<username>, <freekbyte>)
  - o CLSET\_UserRate  
(<username>, <ratio>)
  
  - o CLSET\_UserPassword(<username>, <passwort>)  
  
Setzt für einen User das Passwort.  
  
Parameter:  
  
username - Username des Users  
passwort - Passwort des Users
  
  - o CLSET\_UserPriority(<username>, <priorität>)
-

Setzt die Priorität eines Users.

Parameter:

username - Username des Users  
priorität- Priorität (0<=x<=99)

- o CLSET\_UserPath(<username>,<pfad>)

Setzt den Pfad des Users.

Parameter:

username - Username des Users  
pfad - gültiger DOS-Pfad des Userverzeichnis

- o CLSET\_UserGroup(<username>,<gruppennr> [,<flag>])

Trägt den User in eine Gruppe ein bzw. aus.

Parameter:

username - Username des Users  
gruppennr- Nummer der Gruppe  
flag - optionales Flag; ist es '0', wird der User aus dieser Gruppe ausgetragen. Ist es '1' oder fehlt es gänzlich, wird ein Eintragen in die angegebene Gruppe versucht.

- o CLSET\_UserRealname(<username>,<realname>)

Setzt den Realnamen des Users.

Parameter:

username - Username des Users  
realname - Realname des Users

- o CLSET\_UserAddress(<username>,<adresse>)

Setzt die Adresse des Users.

Parameter:

username - Username des Users  
adresse - Adresse des Users

- o CLSET\_UserTelephone(<username>,<telefon>)

Setzt die Telefonnummer des Users.

Parameter:

---

username - Username des Users  
telefon - Telefonnummer des Users (String)

o CLSET\_UserSysopInfo(<username>,<sysopinfo>)

Setzt die Sysopinfo des Users.

Parameter:

username - Username des Users  
sysopinfo- Informationstext

o CLSET\_UserMessage(<username>,<message>)

Setzt die Userinfo des Users.

Parameter:

username - Username des Users  
unserinfo- Informationstext

o CLSET\_UserCopyTo(<username>,<vertreter>)

Setzt den Vertreter des Users.

Parameter:

username - Username des Users  
vertreter- Vertreter des Users

o CLSET\_UserTerminal(<username>,<terminal>)

Setzt Terminaleinstellung des Users.

Parameter:

username - Username des Users  
terminal - Entweder 'ANSI', 'VT100' oder 'TTY'

o CLSET\_UserLanguage(<username>,<sprache>)

Setzt die Spracheinstellung des Users.

Parameter:

username - Username des Users  
sprache - Sprache des Users (Default: 'DEUTSCH')

o CLSET\_UserKeyboard(<username>,<zeichensatz>)

Setzt den Zeichensatz des Users.

---

Parameter:

username - Username des Users  
zeichensatz - Entweder "IBM" oder "ISO"

o CLSET\_UserTimeout(<username>,<timeout>)

Setzt den Zeitlimit des Users.

Parameter:

username - Username des Users  
timeout - Zeitlimit des Users in Minuten

o CLSET\_UserAccount(<username>,<konto>)

Setzt den Kontostand des Users.

Parameter:

username - Username des Users  
konto - Kontostand des Users

o CLSET\_UserCredit(<username>,<kredit>)

Setzt den erlaubten Kredit des Users.

Parameter:

username - Username des Users  
kredit - Kredit

o CLSET\_UserLines(<username>,<zeilenzahl>)

Setzt die Zeilenzahl-Einstellung des Users.

Parameter:

username - Username des Users  
zeilenzahl - Zeilenzahl des Users

o CLSET\_UserProtocol(<username>,<protokoll>)

Setzt das Übertragungsprotokoll.

Parameter:

username - Username des Users  
protokoll- Übertragungsprotokoll (Default: "ZMODEM")

---

---

o CLSET\_UserMaxKB (<username>, <maxkb>)

Setzt das maximale Haltevolumen in Kilobyte für den Cruncher im Postfach des Users.

Parameter:

username - Username des Users  
maxkb - Maximalvolumen

o CLSET\_UserMaxDays (<username>, <maxdays>)

Setzt die maximale Haltezeit in Tagen für den Cruncher im Postfach des Users.

Parameter:

username - Username des Users  
maxdays - maximale Haltezeit

o CLSET\_UserUploads (<username>, <uploads>)

Setzt die Uploads des Users in Kilobyte.

Parameter:

username - Username des Users  
uploads - Uploads

o CLSET\_UserDownloads (<username>, <downloads>)

Setzt die Downloads des Users in Kilobyte.

Parameter:

username - Username des Users  
downloads- Downloads

o CLSET\_UserFreeFiles (<username>, <freefiles>)

Setzt die freien Files des Users.

Parameter:

username - Username des Users  
freefiles- Anzahl der freien Files.

o CLSET\_UserFreeKByte (<username>, <freekbyte>)

Setzt den Downloadfreiraum des Users.

Parameter:

---

username - Username des Users  
freekbyte- Downloadfreiraum

- o CLSET\_UserRate(<username>,<ratio>)

Setzt die Up-/Downloadratio des Users.

Parameter:

username - Username des Users  
ratio - Up-/Downloadratio (1:x)

### 3.4 Befehle zum Lesen der Userflags

---

Ist das jeweilige Flag gesetzt, liefern die Funktionen '1' zurück, ansonsten '0'.

- o CLIS\_UserInvisible  
(<username>)
- o CLIS\_UserNetworkEnabled  
(<username>)
- o CLIS\_UserAutoEntry  
(<username>)
- o CLIS\_UserAutoRemove  
(<username>)

- o CLIS\_UserInvisible(<username>)

Abfrage, ob der User unsichtbar ist.

- o CLIS\_UserNetworkEnabled(<username>)

Abfrage, ob der User netzberechtigt ist.

- o CLIS\_UserAutoEntry(<username>)

Abfrage, ob der User automatisch eingetragen wurde (unbearbeiteter Userantrag).

- o CLIS\_UserAutoRemove(<username>)

Abfrage, ob der User automatisch entfernt werden soll.

---

### 3.5 Befehle zum Setzen der Userflags

---

Bei den Funktionen dieses Abschnittes sind die zu übergebenden Parameter von Funktion zu Funktion identisch und haben folgenden Inhalt:

Parameter:

username - Name des Users  
flag - '0' bedeutet, das Flag wird gelöscht (FALSE)  
          '1' bedeutet, das Flag wird gesetzt (TRUE)

- o           CLSET\_UserInvisible  
            (<username>,<flag>)
  - o           CLSET\_UserNetworkEnabled  
            (<username>,<flag>)
  - o           CLSET\_UserAutoEntry  
            (<username>,<flag>)
  - o           CLSET\_UserAutoRemove  
            (<username>,<flag>)
- 
- o CLSET\_UserInvisible(<username>,<flag>)  
    Flag, ob der User unsichtbar ist.
  - o CLSET\_UserNetworkEnabled(<username>,<flag>)  
    Flag, ob der User Netzberechtigung hat.
  - o CLSET\_UserAutoEntry(<username>,<flag>)  
    Flag, ob der User automatisch eingetragen wurde.
  - o CLSET\_UserAutoRemove(<username>,<flag>)  
    Flag, ob der User automatisch gelöscht wird.

## 1.5 Befehle für Zugriffe auf Systeme

### 4. Befehle für Zugriff auf Systemdaten

---

## Allgemeine Anmerkungen:

-----

Wenn versucht wird, auf ein nicht existentes System zuzugreifen, wird beim Lesen generell ein Leerstring zurückgegeben, beim Schreiben '0'.

## 4.1

Allgemeine~Befehle

4.2

Befehle~zum~Auslesen~der~Systemdaten

4.3

Befehle~zum~Setzen~der~Systemdaten

4.1 Allgemeine Befehle

-----

o

CL\_SystemSavelist

()

o

CL\_SystemNew

(&lt;systemname&gt;)

o

CL\_SystemDelete

(&lt;systemname&gt;)

o

CLGET\_SystemNumberOf

()

o

CLGET\_SystemName

(&lt;index&gt;)

o

CLIS\_SystemExists

(&lt;systemname&gt;)

o CL\_SystemSavelist ()

Wurde die Systemliste verändert, wird sie bei Aufruf dieser Funktion gespeichert.

o CL\_SystemNew (&lt;systemname&gt;)

Erzeugt ein neues System mit dem Namen <systemname>. Danach sollten mit den CLSET\_SystemX()-Befehlen (siehe Abschnitt

4.3

) alle Daten

ergänzt werden.

o CL\_SystemDelete (&lt;systemname&gt;)

Das System <systemname> wird aus der Systemliste gelöscht.

- o CLGET\_SystemNumberOf()

Diese Funktion gibt die Anzahl aller eingetragenen Systeme zurück.

- o CLGET\_SystemName(<index>)

Nach Aufruf gibt die Funktion den Namen des Systems mit der Nummer <index> zurück.

Beispiel:

```
anzahl=
  CLGET_SystemNumberOf
  () /* Anzahl der Systeme ermitteln */

do x=1 to anzahl
  system.x=CLGET_SystemName(x)
end
```

- o CLIS\_SystemExists(<systemname>)

Liefert '1', wenn das System <systemname> in der Systemliste eingetragen ist, ansonsten '0'.

#### 4.2 Befehle zum Auslesen der Systemdaten

---

- o CLGET\_SystemDomain  
(<systemname>)
  - o CLGET\_SystemRoute  
(<systemname>)
  - o CLGET\_SystemNet  
(<systemname>)
  - o CLGET\_SystemProtocol  
(<systemname>)
  - o CLGET\_SystemPackerIn  
(<systemname>)
  - o CLGET\_SystemPackerOut  
(<systemname>)
  - o CLGET\_SystemGroup  
(<systemname>, <gruppennr>)
  - o CLGET\_SystemPointuser
-

- o (<systemname>)
- o CLGET\_SystemTelephone  
(<systemname>)
- o CLGET\_SystemPriority  
(<systemname>)
- o CLGET\_SystemLastCall  
(<systemname>)
- o CLIS\_SystemAutoEntry  
(<systemname>)
- o CLIS\_SystemMapsAccess  
(<systemname>)

- o CLGET\_SystemDomain(<systemname>)

Diese Funktion gibt die Domain eines eingetragenen Systems zurück.

- o CLGET\_SystemRoute(<systemname>)

Diese Funktion liefert das Routesystem von <systemname>. Bei nicht existentem System <systemname> oder bei aktiviertem PathAlias-/Domainrouting wird ein Leerstring zurückgeliefert.

Beispiel:

```
route=CLGET_SystemRoute(systemname)
if route='' then do
  if ~
    CLIS_SystemExists
    (systemname) then
    say 'Dieses System ist nicht eingetragen!'
  else
    say 'Mail wird per PathAlias/Domainrouting geroutet!'
end
```

- o CLGET\_SystemNet (<systemname>)

Liefert den Netznamen vom System <systemname> im Klartext (JANUS, ZERBERUS, UUCP oder FIDO).

- o CLGET\_SystemProtocol (<systemname>)

Liefert den Namen des für <systemname> eingestellten Transferprotokolls im Klartext.

- o CLGET\_SystemPackerIn (<systemname>)

Liefert den Packer der von <systemname> eingehenden Puffer im Klartext.

o CLGET\_SystemPackerOut (<systemname>)

Liefert den Packer der an <systemname> ausgehenden Puffer im Klartext.

o CLGET\_SystemGroup (<systemname>, <gruppennr>)

Liefert '1', wenn das System <systemname> in die Gruppe mit der Nummer <gruppennr> eingetragen ist, ansonsten '0'.

o CLGET\_SystemPointuser (<systemname>)

Falls es sich bei dem System um einen Point handelt, wird der Name des Pointusers zurückgegeben, ansonsten ein Leerstring.

o CLGET\_SystemTelephone (<systemname>)

Diese Funktion gibt die Telefonnummer des Systems <systemname> zurück.

Beispiel:

```
/* Einlesen der einzelnen Telefonnummern eines Systems in eine
   STEM-Variable */
string=translate(CLGET_SystemTelephone(system),' ','|')
if string~='' then do
  do x=1 to words(string)
    telefon.x=word(string,x)
  end
end
```

o CLGET\_SystemPriority (<systemname>)

Diese Funktion gibt die Priorität des Systems <systemname> zurück.

o CLGET\_SystemLastCall (<systemname>)

Liefert das Datum des letzten Anrufs vom System <systemname> im Format 'DD.MM.YYYY HH:MM:SS'.

o CLIS\_SystemAutoEntry (<systemname>)

Liefert den Rückgabewert '1', wenn das System die Berechtigung zum Autoeintrag neuer Brett hat, ansonsten '0'.

- o CLIS\_SystemMapsAccess (<systemname>)

Liefert den Rückgabewert '1', wenn das System Zugriff auf Maps hat, ansonsten '0'.

#### 4.3 Befehle zum Setzen der Systemdaten

-----

Bei Erfolg wird generell '1' zurückgegeben, bei Mißerfolg '0'.

- o CLSET\_SystemPassword  
(<systemname>, <password>)
  - o CLSET\_SystemDomain  
(<systemname>, <domain>)
  - o CLSET\_SystemRoute  
(<systemname>, <routesystem>)
  - o CLSET\_SystemNet  
(<systemname>, <netzname>)
  - o CLSET\_SystemProtocol  
(<systemname>, <protokoll>)
  - o CLSET\_SystemPackerIn  
(<systemname>, <packer>)
  - o CLSET\_SystemPackerOut  
(<systemname>, <packer>)
  - o CLSET\_SystemGroup  
(<systemname>, <gruppennr> [, <flag>])
  - o CLSET\_SystemPointuser  
(<systemname>, <username>)
  - o CLSET\_SystemTelephone  
(<systemname>, <telefon>)
  - o CLSET\_SystemPriority  
(<systemname>, <priorität>)
  - o CLSET\_SystemAutoEntry  
(<systemname>, <flag>)
  - o CLSET\_SystemMapsAccess  
(<systemname>, <flag>)
-

- o CLSET\_SystemPassword(<systemname>,<passwort>)  
Trägt für das System das Passwort <passwort> ein.
  - o CLSET\_SystemDomain(<systemname>,<domain>)  
Trägt für das System die Domain <domain> ein.
  - o CLSET\_SystemRoute(<systemname>,<routesystem>)  
Trägt für das System das Routesystem <routesystem> ein und schaltet ein evt. eingeschaltetes PathAlias/Domainrouting ab.
  - o CLSET\_SystemNet(<systemname>,<netzname>)  
Setzt den Netznamen des Systems (Default "JANUS").
  - o CLSET\_SystemProtocol(<systemname>,<protokoll>)  
Setzt das Transferprotokoll des Systems (Default "ZMODEM").
  - o CLSET\_SystemPackerIn(<systemname>,<packer>)  
Setzt den Packer für eingehende Puffer vom System (Default "LHA").
  - o CLSET\_SystemPackerOut(<systemname>,<packer>)  
Setzt den Packer für ausgehende Puffer an das System (Default "LHA").
  - o CLSET\_SystemGroup(<systemname>,<gruppennr> [,<flag>])  
Trägt das System in eine Gruppe ein.  
  
Parameter:  
  
systemname - Name des Systems  
gruppennr - Nummer der Gruppe  
flag - Optionaler Parameter; ist es "0", wird das System aus der Gruppe ausgetragen, ist es "1" oder wird weggelassen, wird das System in die Gruppe eingetragen.
  - o CLSET\_SystemPointuser(<systemname>,<username>)  
Setzt einen Pointuser für das System.  
  
Parameter:  
  
systemname - Name des Systems
-

username - Pointuser. Wird hier ein Leerstring übergeben, wird die Verknüpfung gelöscht und das System nicht weiter als Point betrachtet.

o CLSET\_SystemTelephone(<systemname>,<telefon>)

Setzt die Telefonnummer des Systems.

Parameter:

systemname - Name des Systems  
telefon - String mit einer oder mehreren Telefonnummern, bei mehreren durch '|' getrennt

Beispiel:

```
if zaehler>1 then do
  do x=1 to zaehler-1 /* Mehrere Nummern aneinanderhängen */
    telefon=nummer.x'|'
  end
end
telefon=telefon|nummer.zaehler /* Nur ein Port bzw. der letzte */
call CLSET_SystemTelephone(system,telefon)
```

o CLSET\_SystemPriority(<systemname>,<priorität>)

Setzt die Priorität des Systems.

Parameter:

systemname - Name des Systems  
priorität - Priorität des Systems (0<=x<=99)

o CLSET\_SystemAutoEntry(<systemname>,<flag>)

Setzt die Berechtigung zum Autoeintrag von Brettern.

Parameter:

systemname - Name des Systems  
flag - '1' erlaubt Autoeintrag  
      '0' verweigert Autoeintrag

o CLSET\_SystemMapsAccess(<systemname>,<flag>)

Setzt die Berechtigung zum Zugriff auf Maps.

Parameter:

systemname - Name des Systems  
flag - '1' erlaubt Mapszugriff  
      '0' verweigert Mapszugriff

---

## 1.6 Funktionen für Onlinebefehle

### 5. Funktionen für Onlinebefehle

---

Allgemeine Anmerkungen:

---

Alle CLONL\_X()-Befehle liefern bei Erfolg '1' und bei Mißerfolg '0' zurück. Es sollte zumindest bei ersten Aufruf überprüft werden, ob eine '1' zurückgegeben wurde, damit man weiß, ob das ANSI-System korrekt initialisiert wurde.

Beispiel:

```
if ~
    CLONL_ClearScreen
    () then do
    say "Can't init ANSI-System!"
    exit 10
end

o
    CL_Edit
    (<filename> [,<flag>])
o
    CLONL_GotoXY
    (<x>,<y>)
o
    CLONL_Normal
    ()
o
    CLONL_DeleteEOL
    ()
o
    CLONL_DeleteLine
    ()
o
    CLONL_InsertLine
    ()
o
    CLONL_BlueMode
    ()
o
    CLONL_Color
    (<color>,<bcolor>,<style>)
o
    CLONL_BlueRead
    (<länge> [,<vorgabe>])
o
    CLONL_Wait
    ()
o
    CLONL_DoCommand
    (<befehl> [,<parameter>])
```

---

- o CLONL\_SendMail  
(<absender>, <empfänger>, <betreff>, <filename>)
- o CLGET\_Key()
  - o CLONL\_Transfer  
(<filename> [, <sendfilename>])
- o CLONL\_Receive  
(<port>)
- o CLIS\_KeyPressed  
( )
- o CLONL\_ShortMenu  
(<eintrag> {, <einträge>})
- o CLONL\_ShortMenuVertical  
(<eintrag> {, <einträge>})
- o CLONL\_SetObscure  
(<flag>)
- o CLONL\_ClearScreen  
( )
- o CLONL\_PrintFile  
(<filename>)
- o CLONL\_SetEnv  
(<port>)

- o CL\_Edit (<filename> [, <flag>])

Ruft den Editor auf.

Parameter:

filename - Pfad und Name der Datei <filename>  
online - Optionaler Parameter; ist er angegeben, wird  
der Online-Editor anstelle des Offline-Editors  
benutzt.

Beispiel:

```
call CL_Edit('T:test',) /* Online-Editor benutzen */
```

- o CLONL\_GotoXY (<x>, <y>)

Setzt den Cursor auf die Position x,y.

Parameter:

---

x - Nummer der Spalte  
y - Nummer der Zeile

Beispiel:

```
call CLONL_GotoXY(1,1) /* Cursor in die linke obere Ecke */
```

o CLONL\_Normal()

Schaltet die Textausgabe auf ANSI-Style "Normal" zurück. Das Resultat ist von Terminal zu Terminal verschieden; üblicherweise ist es Schriftfarbe weiß, Hintergrundfarbe Schwarz, kein Style.

o CLONL\_DeleteEOL()

Per ANSI-Steuersequenz wird von der aktuellen Cursorposition bis zum Ende der Zeile alles gelöscht.

o CLONL\_DeleteLine()

Per ANSI-Steuersequenz wird die aktuelle Zeile vollständig gelöscht, nachfolgende Zeilen rücken auf.

o CLONL\_InsertLine()

Per ANSI-Steuersequenz wird an der aktuellen Position des Cursors eine Zeile eingefügt und die nachfolgende Zeilen um eine Zeile nach unten verschoben.

o CLONL\_BlueMode()

Es wird auf den 'Blue'-Mode umgestellt. Hierbei handelt es sich um einen besonders markanten Eingabe- und Titelmodus.

o CLONL\_Color(<color>,<bcolor>,<style>)

Die Cursorfarbe wird zur weiteren Ausgabe umgestellt, ein Schriftstil wird benutzt.

Parameter:

color - Schriftfarbe (BLACK, RED, GREEN, YELLOW, BLUE, PINK, CYAN oder WHITE)

bcolor- Hintergrundfarbe (BLACK, RED, GREEN, YELLOW, BLUE, PINK, CYAN oder WHITE)

style - Schriftstil (NORMAL, BOLD, KURSIV, UNDERLINE oder INVERSE)

o CLONL\_BlueRead(<länge> [,<vorgabe>]

Über eine blaue Eingabezeile wird ein String-Gadget simuliert. Exis-

tiert eine Vorgabe, so steht diese bereits im Feld. Nach einer Bestätigung der Eingabe der 'Return'-Taste wird der Inhalt des Feldes zurückgegeben.

Parameter:

länge - Länge der Eingabezeile in Zeichen  
vorgabe - Text, der als Default in der Eingabezeile stehen soll

o CLONL\_Wait()

Es wird eine Meldung "<Bitte Taste drücken>" ausgegeben und auf einen Tastendruck gewartet. Wird eine '1' zurückgegeben, wurde eine Taste gedrückt. Bei einem Rückgabewert '0' sollte sich das Programm auf jeden Fall vollständig beenden, da der User dann CTRL-C/Q/X gedrückt hat oder der Carrier verloren wurde.

o CLONL\_DoCommand(<befehl> [,<parameter>])

Ruft einen in den Shellprefs eingetragenen Onlinebefehl auf, so, als würde der User den Befehl selbst aufrufen.

Parameter:

befehl - Onlinebefehl  
parameter - Optionaler Parameter; beinhaltet einen String, der an den Befehl als Parameter übergeben wird.

o CLONL\_SendMail(<absender>,<empfänger>,<betreff>,<filename>)

Verschickt eine Textnachricht. Siehe auch CL\_SendMailFull().

Parameter:

absender - Absender der Nachricht  
empfänger - Empfänger der Nachricht  
betreff - Betreff der Nachricht  
filename - Pfad und Name des zu verschickenden Files.

o CLGET\_Key()

Wartet auf einen Tastendruck und gibt sie zurück. Ein String '-1' bedeutet, daß ein Abbruch stattgefunden hat oder der Carrier verloren wurde; in beiden Fällen sollte sich das ARexx-Skript umgehend selbst beenden.

Bei gedrückten Cursortasten werden folgende Werte zurückgegeben:

Cursor nach oben : "-UP"  
Cursor nach unten : "-DOWN"  
Cursor nach links : "-LEFT"  
Cursor nach rechts: "-RIGHT"

Beispiel:

```

menu:
  say '[1] Nachricht editieren'
  say '[2] Nachricht verschicken'
  say '[E] Programm verlassen'
  say ''
loop:
  taste=CLGET_Key()
  select
    when taste='1' then call edit
    when taste='2' then call send
    when (taste='E')|(taste='-1') then exit
    otherwise call loop
  end
call menu

```

o CLONL\_Transfer(<filename> [,<sendfilename>]

Verschickt ein File an den User. Bei dieser Funktion kann es Probleme geben, wenn sie in ihrem System XPR-Treiber-Protokolle anstelle der Connectline Built-in-Übertragungsprotokolle verwenden.

Parameter:

filename - Name des zu übertragenden Files  
 sendfilename- Optionaler Parameter; ist er angegeben, wird das File mit diesem Namen verschickt.

Beispiel:

```

if ~CLONL_Transfer('JANUS:'||point||'/'||point||'.PRE') then
  say 'Transfer fehlgeschlagen!'
else
  say 'Transfer erfolgreich!'

```

o CLONL\_Receive(<port>)

Empfängt ein File auf Port <port> und gibt den kompletten Pfad zurück, bzw. einen Leerstring bei fehlgeschlagener Übertragung. Werden mehrere Files empfangen, so werden die Filenamen durch Leerzeichen getrennt in einem String zurückgegeben. Die empfangenen Files sollten auf jeden Fall entsorgt werden, da sie sonst vom Onlinebefehl SENDEN als Bimodem-Übertragungsfiles behandelt werden.

Beispiel:

```

string=CLONL_Receive(port)
if string~='' then do
  do x=1 to words(string)
    rename(word(string,x),'T:'||word(string,words(translate(string,
' ','/'))))
  end
end
else say 'Übertragung fehlgeschlagen!'

```

- o CLIS\_KeyPressed()

Liefert '1', wenn eine Taste gedrückt wurde, ansonsten '0'.

- o CLONL\_ShortMenu(<eintrag> {,<einträge>})

Es wird ein kleines einzeiliges Menü aus den Einträgen aufgebaut. Die maximale Anzahl der Parameter liegt bei 20. Der erste Buchstabe eines jeden Eintrages wird als Hotkey verwendet. Der Hotkey des ausgewählten Eintrages wird zurückgegeben.

- o CLONL\_ShortMenuVertical(<eintrag> {,<einträge>})

Es wird ein kleines vertikales Menü aus den Einträgen aufgebaut. Die maximale Anzahl der Parameter liegt bei 20. Der erste Buchstabe eines jeden Eintrages wird als Hotkey verwendet. Der Hotkey des ausgewählten Eintrages wird zurückgegeben.

- o CLONL\_SetObscure(<flag>)

Setzt den Obscure-Modus. Ist das Flag gesetzt, ist ein Mitlesen auf der Console nicht mehr möglich.

- o CLONL\_ClearScreen()

Per ANSI-Steuersequenz wird der Bildschirm gelöscht.

- o CLONL\_PrintFile(<filename>)

Gibt das File <filename> als Text aus.

- o CLONL\_SetEnv(<port>)

Setzt die lokalen ENV-Variablen des Ports <port> als lokale ENV-Variablen des aktuellen ARexx-Prozesses. Dies erlaubt das Aufrufen von Online-Befehlen aus einem ARexx-Skript heraus.

Beispiel:

```
/* Ruft 'TopUser' aus einem ARexx-Skript auf */
```

```
port=arg(1)
call CLONL_SetEnv(port)
address command 'CONNECTLINE:Commands/TopUser'
exit
```

## 1.7 Befehle für Zugriff auf Nachrichten

### 6. Befehle für Zugriffe auf Nachrichten

---

#### 6.1

Befehle~zum~Lesen~von~Nachrichtendaten

#### 6.2

Befehle~zum~Setzen~von~Nachrichtendaten

6.1 Befehle zum Lesen von Nachrichtendaten

---

- o CLGET\_MsgNumberOf  
(<brettname>)
  - o CLGET\_MsgSubject  
(<brettname>,<nr>)
  - o CLGET\_MsgFilename  
(<brettname>,<nr>)
  - o CLGET\_MsgShortComment  
(<brettname>,<nr>)
  - o CLGET\_MsgFrom  
(<brettname>,<nr>)
  - o CLGET\_MsgMsgID  
(<brettname>,<nr>)
  - o CLGET\_MsgReadcount  
(<brettname>,<nr>)
  - o CLGET\_MsgIncomeDate  
(<brettname>,<nr>)
  - o CLGET\_MsgSendDate  
(<brettname>,<nr>)
  - o CLIS\_MsgBinary  
(<brettname>,<nr>)
  - o CLGET\_MsgBody  
(<brettname>,<nr>,<filename>)
  - o CLGET\_MsgBodyLength  
(<brettname>,<nr>)
  - o CLGET\_MsgComment  
(<brettname>,<nr>,<filename>)
  - o CLGET\_MsgCommentLength  
(<brettname>,<nr>)
-

- o CLGET\_MsgNumberOf (<brettname>)  
Liefert die Anzahl der Nachrichten im Brett <brettname> zurück.  
Existiert das Brett nicht oder konnte der Index nicht gelockt werden, wird eine '-1' zurückgegeben.
  - o CLGET\_MsgSubject (<brettname>, <nr>)  
Liefert den Betreff von Nachricht Nr. <nr> im Brett <brettname>.
  - o CLGET\_MsgFilename (<brettname>, <nr>)  
Liefert den Filenamen von Nachricht Nr. <nr> im Brett <brettname>.
  - o CLGET\_MsgShortComment (<brettname>, <nr>)  
Liefert den Kurzkomentar von Nachricht Nr. <nr> im Brett <brettname>.
  - o CLGET\_MsgFrom (<brettname>, <nr>)  
Liefert den Absender von Nachricht Nr. <nr> im Brett <brettname>.
  - o CLGET\_MsgMsgID (<brettname>, <nr>)  
Liefert die MsgID von Nachricht Nr. <nr> im Brett <brettname>.
  - o CLGET\_MsgReadcount (<brettname>, <nr>)  
Liefert den Zugriffszähler von Nachricht Nr. <nr> im Brett <brettname>.
  - o CLGET\_MsgIncomeDate (<brettname>, <nr>)  
Liefert das Einsortierdatum von Nachricht Nr. <nr> im Brett <brettname>.
  - o CLGET\_MsgSendDate (<brettname>, <nr>)  
Liefert das Absendedatum von Nachricht Nr. <nr> im Brett <brettname>.
  - o CLIS\_MsgBinary (<brettname>, <nr>)  
Liefert '1', wenn die Nachricht Nr. <nr> im Brett <brettname> ein File ist, ansonsten '0'.
  - o CLGET\_MsgBody (<brettname>, <nr>, <filename>)
-

Diese Funktion lagert den Body der Nachricht Nr. <nr> im Brett <brettname> mit dem Filenamen <filename> aus.

- o CLGET\_MsgBodyLength(<brettname>,<nr>)

Diese Funktion gibt die Länge des Mailbodies der Nachricht Nr. <nr> im Brett <brettname> zurück.

- o CLGET\_MsgComment(<brettname>,<nr>,<filename>)

Diese Funktion dient dem Auslagern des langen Kommentars (KOM:) der Nachricht Nr. <nr> des Brettes <brettname> unter dem Filename <filename>.

- o CLGET\_MsgCommentLength(<brettname>,<nr>)

Diese Funktion gibt die Länge des langen Kommentars (KOM:) der Nachricht Nr. <nr> des Brettes <brettname> zurück.

## 6.2 Befehle zum Setzen von Nachrichtendaten

---

- o  
    CLSET\_MsgSubject  
    (<brettname>,<nr>,<betreff>)
- o  
    CLSET\_MsgShortComment  
    (<brettname>,<nr>,<kommentar>)
- o  
    CLSET\_MsgDelete  
    (<brettname>,<nr>)

- o CLSET\_MsgSubject(<brettname>,<nr>,<betreff>)

Setzt den Betreff von Nachricht Nr. <nr> im Brett <brettname> auf <betreff>.

- o CLSET\_MsgShortComment(<brettname>,<nr>,<kommentar>)

Setzt den Kurzkomentar von Nachricht Nr. <nr> im Brett <brettname> auf <kommentar>.

- o CLSET\_MsgDelete(<brettname>,<nr>)

Setzt das Löschflag von Nachricht Nr. <nr> im Brett <brettname>. Die

---

Nachricht wird beim nächsten Crunchen des Brettes vollständig entfernt.

## 1.8 Befehle für Zugriffe auf die Ports

### 7. Befehle für Zugriffe auf die Ports

---

Allgemeine Anmerkungen:

---

Diese Funktionen benötigen als Übergabeparameter die Nummer des Ports, von welchem aus das Skript gestartet wurde. Diese erhalten sie entweder beim Aufruf als Parameter oder über den ARexx-Port des CL-PORTx.

Ist niemand eingeloggt, geben die Funktionen - solange nicht anders vermerkt - einen Leerstring zurück.

- o CLGET\_PortUsername  
(<port>)
  - o CLGET\_PortCurrentboard  
(<port>)
  - o CLGET\_PortTerminal  
(<port>)
  - o CLGET\_PortLines  
(<port>)
- 
- o CLGET\_PortUsername (<port>)  
  
Liefert den Usernamen des eingeloggten Users.
  - o CLGET\_PortCurrentboard (<port>)  
  
Liefert den Namen des Brettes, in welchem sich der User gerade aufhält.
  - o CLGET\_PortTerminal (<port>)  
  
Liefert die aktuelle Terminaleinstellung des Users im Klartext ("ANSI", "VT100" oder "TTY").
  - o CLGET\_PortLines (<port>)  
  
Liefert die aktuelle maximale Zeilenzahl des Users. Ist niemand ein-
-

geloggt, wird eine '-1' zurückgegeben.

## 1.9 Befehle zum Auslesen von Defaultpfaden

### 8. Befehle zum Auslesen von Defaultpfaden

- o
  - o CLGET\_PathJANUS()
  - o
    - o CLGET\_PathZERBERUS()
- o CLGET\_PathSpool()
- o CLGET\_PathTrash()
- o CLGET\_PathBoard()
- o CLGET\_PathUser()

#### o CLGET\_PathJANUS()

Liefert den in den JANUS-Prefs eingestellten Pfad des JANUS-Verzeichnisses zurück. Beim Aufruf dieser Funktion muß Connectline laufen.

Beispiel:

```
/* Abfragen, ob Connectline zur Zeit läuft */
if ~show('P','CL-SERVER') then do
  say 'Bitte erst Connectline 5.0 starten!'
  exit
end
JANUS=CLGET_PathJANUS()
```

#### o CLGET\_PathZERBERUS()

Liefert den in den Z38-Prefs eingestellten Pfad des ZERBERUS-Verzeichnisses zurück. Beim Aufruf dieser Funktion muß Connectline laufen.

Beispiel:

```
/* Abfragen, ob Connectline zur Zeit läuft */
if ~show('P','CL-SERVER') then do
  say 'Bitte erst Connectline 5.0 starten!'
  exit
```

- ```
end
ZERBERUS=CLGET_PathZERBERUS()
```
- o CLGET\_PathSpool()  
Liefert den Pfad des JANUS-Spoolverzeichnis zurück.
  - o CLGET\_PathTrash()  
Liefert den Pfad des JANUS-Trashverzeichnis zurück.
  - o CLGET\_PathBoard()  
Liefert den in der Hauptkonfiguration eingestellten Defaultpfad der  
Bretter zurück.
  - o CLGET\_PathUser()  
Liefert den in der Hauptkonfiguration eingestellten Defaultpfad der  
Userverzeichnisse zurück.

## 1.10 Index

Befehlsübersicht~des~Servers~und~der~Ports  
Übersicht der cl\_rexx-Funktionen

CLInfo()

CL\_BoardDelete

CL\_BoardNew

CL\_Edit

CL\_GetText

CL\_LogAdd

CL\_LogDelete

CL\_LogRename

CL\_PutStr

CL\_SaveBoardlist

CL\_SendMailFull

---

CL\_SystemDelete  
CL\_SystemNew  
CL\_SystemSavelist  
CL\_UserNew  
CLGET\_BoardAbout  
CLGET\_BoardAdmin  
CLGET\_BoardBoxList  
CLGET\_BoardCensor  
CLGET\_BoardFTSName  
CLGET\_BoardHoldDays  
CLGET\_BoardHoldKB  
CLGET\_BoardLastWrite  
CLGET\_BoardList  
CLGET\_BoardMaxNet  
CLGET\_BoardNetCount  
CLGET\_BoardNewsgroupName  
CLGET\_BoardNext  
CLGET\_BoardNumberOf  
CLGET\_BoardPath  
CLGET\_BoardPrefix  
CLGET\_BoardPrevious  
CLGET\_BoardPri  
CLGET\_BoardValidateBox  
CLGET\_BoardValidateUser  
CLGET\_BoardWritePri  
CLGET\_FileRequest  
CLGET\_FrontScreenName  
CLGET\_GroupName  
CLGet\_GroupName

---

---

CLGET\_Key ()  
CLGET\_MsgBody  
CLGET\_MsgBodyLength  
CLGET\_MsgComment  
CLGET\_MsgCommentLength  
CLGET\_MsgFilename  
CLGET\_MsgFrom  
CLGET\_MsgIncomeDate  
CLGET\_MsgMsgID  
CLGET\_MsgNumberOf  
CLGET\_MsgReadcount  
CLGET\_MsgSendDate  
CLGET\_MsgShortComment  
CLGET\_MsgSubject  
CLGET\_OwnSystemAddress  
CLGET\_OwnSystemName  
CLGET\_PackerList  
CLGET\_PathBoard  
CLGET\_PathJANUS ()  
CLGET\_PathSpool  
CLGET\_PathTrash  
CLGET\_PathUser  
CLGET\_PathZERBERUS  
CLGET\_PortCurrentboard  
CLGET\_PortLines  
CLGET\_PortTerminal  
CLGET\_PortUsername  
CLGET\_SystemDomain

---

---

CLGET\_SystemGroup  
CLGET\_SystemLastCall  
CLGET\_SystemName  
CLGET\_SystemNet  
CLGET\_SystemNumberOf  
CLGET\_SystemPackerIn  
CLGET\_SystemPackerOut  
CLGET\_SystemPointuser  
CLGET\_SystemPriority  
CLGET\_SystemProtocol  
CLGET\_SystemRoute  
CLGET\_SystemTelephone  
CLGET\_TransferList  
CLGET\_UserAddress  
CLGET\_UserCopyTo  
CLGET\_UserCredit  
CLGET\_UserDownloads  
CLGET\_UserFreeFiles  
CLGET\_UserFreeKByte  
CLGET\_UserKeyboard  
CLGET\_UserLanguage  
CLGET\_UserLastCall  
CLGET\_UserLines  
CLGET\_UserMaxDays  
CLGET\_UserMaxKB  
CLGET\_UserMessage  
CLGET\_Username  
CLGET\_UserNumberOf  
CLGET\_UserNumberOfCalls

---

---

CLGET\_UserPath  
CLGET\_UserPointname  
CLGET\_UserPriority  
CLGET\_UserProtocol  
CLGET\_UserRate  
CLGET\_UserRealname  
CLGET\_UserSysopInfo  
CLGET\_UserTelephone  
CLGET\_UserTerminal  
CLGET\_UserTimeout  
CLGET\_UserUploads  
CLIS\_BoardAutoEntry  
CLIS\_BoardAutoRemove  
CLIS\_BoardBin  
CLIS\_BoardDLRatio  
CLIS\_BoardExists  
CLIS\_BoardExternal  
CLIS\_BoardGroup  
CLIS\_BoardMapsAllowed  
CLIS\_BoardNoScan  
CLIS\_BoardRealname  
CLIS\_BoardRouteOnly  
CLIS\_BoardText  
CLIS\_KeyPressed  
CLIS\_MsgBinary  
CLIS\_SystemAutoEntry  
CLIS\_SystemExists  
CLIS\_SystemMapsAccess

---

---

CLIS\_UserAutoEntry  
CLIS\_UserAutoRemove  
CLIS\_UserExists  
CLIS\_UserGroup  
CLIS\_UserInvisible  
CLIS\_UserNetworkEnabled  
CLIS\_UserPoint  
CLSET\_BoardAbout  
CLSET\_BoardAddBox  
CLSET\_BoardAdmin  
CLSET\_BoardAutoEntry  
CLSET\_BoardAutoRemove  
CLSET\_BoardBin  
CLSET\_BoardCensor  
CLSET\_BoardDelBox  
CLSET\_BoardDLRatio  
CLSET\_BoardExternal  
CLSET\_BoardFTSName  
CLSET\_BoardGroup  
CLSET\_BoardHoldDays  
CLSET\_BoardHoldKB  
CLSET\_BoardMaxNet  
CLSET\_BoardNewsgroupName  
CLSET\_BoardNoScan  
CLSET\_BoardPri  
CLSET\_BoardRealname  
CLSET\_BoardRouteOnly  
CLSET\_BoardText  
CLSET\_BoardWritePri

---

---

CLSET\_MsgDelete  
CLSET\_MsgShortComment  
CLSET\_MsgSubject  
CLSET\_SystemAutoEntry  
CLSET\_SystemDomain  
CLSET\_SystemGroup  
CLSET\_SystemMapsAccess  
CLSET\_SystemNet  
CLSET\_SystemPackerIn  
CLSET\_SystemPackerOut  
CLSET\_SystemPassword  
CLSET\_SystemPointuser  
CLSET\_SystemPriority  
CLSET\_SystemProtocol  
CLSET\_SystemRoute  
CLSET\_SystemTelephone  
CLSET\_UserAccount  
CLSET\_UserAddress  
CLSET\_UserAutoEntry  
CLSET\_UserAutoRemove  
CLSET\_UserCopyTo  
CLSET\_UserCredit  
CLSET\_UserDownloads  
CLSET\_UserFreeFiles  
CLSET\_UserFreeKByte  
CLSET\_UserGroup  
CLSET\_UserInvisible  
CLSET\_UserKeyboard

---

---

CLSET\_UserLanguage  
CLSET\_UserLines  
CLSET\_UserMaxDays  
CLSET\_UserMaxKB  
CLSET\_UserMessage  
CLSET\_UserNetworkEnabled  
CLSET\_UserPassword  
CLSET\_UserPath  
CLSET\_UserPriority  
CLSET\_UserProtocol  
CLSET\_UserRate  
CLSET\_UserRealname  
CLSET\_UserSysopInfo  
CLSET\_UserTelephone  
CLSET\_UserTerminal  
CLSET\_UserTimeout  
CLSET\_UserUploads  
CLONL\_BlueMode  
CLONL\_BlueRead  
CLONL\_ClearScreen  
CLONL\_Color  
CLONL\_DeleteEOL  
CLONL\_DeleteLine  
CLONL\_DoCommand  
CLONL\_GotoXY  
CLONL\_InsertLine  
CLONL\_Normal  
CLONL\_PrintFile  
CLONL\_Receive

---

```
CLONL_SendMail
CLONL_SetEnv
CLONL_SetObscure
CLONL_ShortMenu
CLONL_ShortMenuVertical
CLONL_Transfer
CLONL_Wait
```

## 1.11 ARexx-Kommandos des CL-SERVER

Die ARexx-Ports von Connectline 5.0

Zusätzlich zu den  
Funktionen `~der~cl_rexx.library`  
stehen Ihnen noch die  
ARexx-Ports von CL-PORTx und des CL-SERVERs zur Verfügung.

### 1. Die ARexx-Schnittstelle des Ports

Der ARexx-Port der einzelnen Ports ist erreichbar unter dem Portnamen CL-PORTx, wobei x hier der jeweiligen Portnummer entspricht ( $1 \leq x \leq 9$ ). Die Telnetports sind nicht mittels ARexx erreichbar, hierfür ist der jeweils benutzte telnetd zuständig.

#### o GETPORTENV <Name>

Liefert den Inhalt der lokalen ENV-Variablen <Name> des Ports zurück.  
Folgende ENV-Variablen sind unter ARexx sinnvoll nutzbar:

```
- CHARSET      (Zeichensatz)
- CURRENTBOARD (aktuelles Brett)
- HOMEDIR      (Pfad des Userverzeichnis)
- LANGUAGE     (Sprache)
- LINES        (Zeilenzahl)
- LOGINPORT    (Port)
- LOGINTIME    (Zeitpunkt des Logins, internes Format)
- PRI          (Priorität des Users)
- TIMEOUT      (restliche Onlinezeit des Users)
- TERMINAL     (Terminalemulation)
- USERNAME     (Username)
- USERREALNAME (Realname)
```

#### o CANCEL

User wird ausgeloggt, Port beendet.

## 2. Die ARexx-Schnittstelle des Servers

Der Name des ARexx-Ports des Servers lautet CL-SERVER.

### o ABORT <Port>

Läßt den Port mit der Nummer <Port> umgehend auflegen.

### o ALLOW\_NETCALL <ON|OFF>

Schaltet das Gadget 'Netcall erlaubt' im Hauptfenster~des~Servers an oder aus.

### o ANSWER <Port>

Läßt den angegebenen Port abheben, als ob ein Anruf eingegangen wäre.

### o BUSY <Port>

Schaltet den angegebenen Port auf besetzt.

### o MESSAGE <Port> <Message>

Verschickt eine Onlinenachricht an den Port, welche dem darauf befindlichen User sofort angezeigt wird.

### o MODEMINIT <Port>

Löst einen Modeminit auf Port <Port> aus.

### o NETCALL <Port> <System> <Retries>

Löst einen Netcall auf Port <Port> bei dem System <System> aus, mit insgesamt <Retries> Anwahlversuchen.

### o OFFLINE <Port>

Schaltet den angegebenen Port offline.

### o ONLINE <Port>

Schaltet den angegebenen Port online.

---

- o PORT\_STATUS <Port>

Dieses Kommando gibt den aktuellen Status von Port <Port> zurück, welcher in der Portübersicht angezeigt wird.

- o PORT\_USER <Port>

Gibt den Usernamen des auf Port <Port> befindlichen Users zurück.

- o SAVE

Speichert die aktuellen Einstellungen des Servers.

---