http://www.bintec.de

RELEASE NOTE

V!CAS
September 24, 1998

New System Software:
Release 4.8 Revision 6

Important Info:

For V!CAS Software Release 4.8 Revision 6 means a functional shift from router with a / b adapter to a PABX device with router functions. This new software concept encloses a series of changes, which are described in this Release Note. If you prefer to keep your V!CAS working with the old router concept, we recommend not to update to Release 4.8 Revision 6 and leave Release 4.8 Revision 3 the current software version of your V!CAS.

First of all you must take notice of the special upgrading procedure, which is necessary with this Software Release. Upgrading is described in detail on page 4 of this Release Note.

The same it is important that a new BRICKware must be installed with Software Release 4.8 Revision 6 to support the PABX functionalities. The BRICKware for BinGO! Plus / Professional (current version 4.8.5) can be retrieved from BinTec's FTP Server at http://www.bintec.de and must be installed together with Software Release 4.8 Revision 6 for V!CAS.

This Software Release 4.8 Revision 6 supports only the DSS1 (Euro-ISDN) D-channel protocol and the Priority Voice feature of the V!CAS gets lost. Priority Voice will be included again in a future release.

With the PABX functionality a user concept is introduced, which is described under The PABX User Concept on page 9.

All Changes that result from this new Software Release are described in the section Changes on page 30.

Contents:

Because of the changed functionality introduced to the V!CAS with Software Release 4.8 Revision 6, we also start a new chapter of documentation for this product with this Release.

Release Notes concerning the V!CAS up to Software Release 4.8 Revision 3 you can find on BinTec's FTP Server at http://www.bintec.de on the page Past Versions of Software and Documentation in the V!CAS section.

Upgrading System Software

1. Retrieve the current system software image from BinTec's WWW server at http://www.bintec.de.

2. Before you now upgrade the V!CAS you must go through the following procedure to make a backup copy of your old configuration and to manage the changes in configuration, which get necessary with the new Software Release 4.8.6. The following commands must be entered as described below from the SNMP shell via a remote host (i.e. using telnet, minipad, or isdnlogin):
   - First you must make a backup of your old configuration for the case of a fall back recovery with the following command:
     `cmd=save path=boot.old`
   - Then to prepare for installing Software Release 4.8.6 the original configuration must be modified by renaming it:
     `cmd=move path=boot pathnew=boot.org`
   - Additionally the configuration of the ISDN numbers in the original configuration must be deleted by deleting the *isdnDispatchTable*. Enter:

```
cmd=delete object=isdnDispatchTable path=boot.org
```

3. Now you can upgrade the V!CAS with the **update** command from the SNMP shell via a remote host (i.e. using telnet, minipad, or isdnlogin) or by using the **BOOTmonitor** if you are logged in directly on the console after a reboot. Notice, when using the **BOOTmonitor**, you have to select the first update option ("Update Flash ROM").
   Information on using the BOOTmonitor can be found in the *V!CAS User's Guide* under *Firmware Upgrades*.
4. After you have updated your V!CAS the new PABX tables must be written to the original configuration and afterwards reloaded with the following commands from the SNMP shell:

```
cmd=save path=boot.org object=pabxusertable
cmd=save path=boot.org object=pabxtrunktable
cmd=save path=boot.org object=pabxtrunkprefixtable
cmd=save path=boot.org object=pabxextensiontable
cmd=load path=boot.org
```

After that save the new configuration with:
`cmd=save`

5. Now you can start to configure the ISDN numbers via the Setup Tool as noted in the section Changes under Call Answering on page 32.
6. Once you've installed Release 4.8 Revision 6 you may want to retrieve the latest documentation (in Adobe's PDF format), which is also available from BinTec's FTP server at the address noted above.

   Note: When upgrading system software to Software Release 4.8.6, it is absolutely necessary that you use the most current version of *BRICKware for Windows for BinGO! Plus/Professional* (Rel. 4.8 Rev. 5, which can be used as well for V!CAS). It can be retrieved from BinTec's FTP server.

   Also pay attention to the information on the next page.

Info:

Performing a software update on a running system (via the **update** command) currently requires that a contiguous block of free memory, ≥ the size of the new software image, is available.

To verify enough memory space is available use the **show mem** command and note the output of the "largest block" field.

To maximize free memory two options are available.

- Perform the update immediately after rebooting the system. This ensures that memory has been defragmented.
- Temporarily reduce the size of your configuration file by deactivating memory intensive software options such as OSPF or IPX.

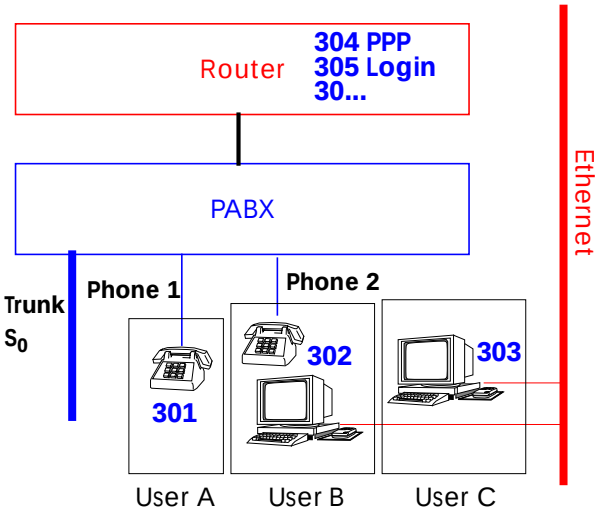
Note that you can always perform an update using the BOOTmonitor. The internal procedure of performing software upgrades on the BRICK is currently being optimized and a change is planned for a future release.

The Voice Data Product Line

With Release 4.8.6 V!CAS will also be moved to the BinTec product line (up to now BinGO! Plus/Professional) that was enhanced by the PABX functionality. This concept means integrating voice into the router product and is a further step to "Integrated Services Networking".

Combining the router functions with PABX allows an easy and cost-effective implementation of many new applications like e.g. Computer Telephony Integration (CTI).

The PABX part of the device is connected to the ISDN network and on the other hand to different terminals (phones, computer applicatons,...) ..



This shift in functionality also has influence on the internal concept of the products. Installing the new Release 4.8.6 means for V!CAS that it is no longer a router with a/b adapter, but must be considered as PABX with router. Changes that could not be avoided affect for example the routing of ISDN calls to the subsystems and the ISDN stacks.

To meet security necessities an user concept is introduced together with the PABX concept. This user concept includes that

extension numbers are related to single users and also terminals are configured for the respective users.
Notice: Throughout this Document the expression **Phone 1/2** is used equivalent to **Phone A/B**, what is the labelling of the POTS sockets on the V!CAS.

What's New in Release 4.8.6

Release 4.8 Revision 6: Released: 21.09.98

Features: | Bugfixes: | Detailed Description:

Features

New Brickware for V!CAS

There is a new separate BRICKware for V!CAS, which is the *BRICKware for BinGO! Plus/Professional*. In its current version 4.8.5 it must be retrieved from BinTecs FTP server at http://www.bintec.de. (Don't forget to download the user documentation for this new Brickware, too.)

When you update your V!CAS with the new Release 4.8.6, you also have to install the new *BRICKware for BinGO! Plus/Professional*, because the new BRICKware includes new functions like the user concept, which are necessary for making use of PABX.

On installing the BRICKware you are quoted for a user and a password. These entries have to correspond to the user configuration on the V!CAS and the later configurations of the CAPI/TAPI applications (also see the chapter Remote CAPI and Remote TAPI in the user documentation for BRICKware for Windows).

The PABX User Concept

Security issues like accessing remote CAPI and remote TAPI via the Ethernet made it necessary to introduce a user concept together with the PABX functions.

User concept means that the PABX extension numbers (MSNs) can via their destination service be assigned to users. TAPI and CAPI applications login with a user and then can see only the calls for the defined user.

A more detailed description you can find at the end of the Setup Tool paragraph.

The configuration extension-user-application is made in the pabxExtension Table respectively in the Setup Tool and is described below.

Setup Tool

Instead of POTS you will find the new menu PABX in the protocol section of the V!CAS Setup Tool, which is described following. The changes in configuration that are made necessary are described in the section Changes.

PABX →

From this menu you can configure phone numbers, users, etc. for the internal PABX (private branch exchange) of your V!CAS.

```
V!CAS Setup Tool                          BinTec Communications GmbH
[PABX]: PABX Configuration                                    vicas

                        Static Settings
                        Extensions
                        User

                        EXIT

Press <Ctrl-n>, <Ctrl-p> to scroll through menu items, <Return> to enter
```

This menu contains three submenus:

STATIC SETTINGS contains the Dial Procedure and TAPI server port settings.

EXTENSIONS lists all extensions defined so far and lets you add new extensions.

USER lists all users defined so far and lets you add new users.

Select **EXIT** to return to the main menu.

PABX → **STATIC SETTINGS** →

The PABX Static Settings menu lets you configure the Dial Procedure and TAPI server port.

```
V!CAS Setup Tool                                BinTec Communications GmbH
[PABX] [STATIC]: PABX Static Settings                                vicas

        Dial Procedure                  Prefix # for internal calls
        Remote TAPI Server Port         6001

                SAVE                                    CANCEL

Use <Space> to select
```

Dial Procedure = This field defines two things: Which prefix is used for internal calls (i.e. for calls between the two POTS (Phone) ports), and which prefix is used for external calls.
There are two possible values:

Value	Internal Prefix	External Prefix
Prefix **#** for **in**ternal calls	#	(none)
Prefix **0** for **ex**ternal calls	(none)	0

The default value (Prefix # for internal calls) means, that internal calls begin with a #, and external calls do not have a special dial prefix.

Remote TAPI Server Port = The TCP port number to use for TAPI connections. Default value: 6001.

PABX → EXTENSIONS →

This menu contains a list of all extensions defined so far. Initially this list will contain three entries, which ensure that *voice* calls will be routed to both Phone ports, and *data* calls will be routed to the *isdnlogin* service.

```
V!CAS Setup Tool                                  BinTec Communications GmbH
[PABX] [EXTENSION]: Configure PABX Extensions                          vicas

   Extension          User            Destination
                                      isdnlogin
                      default         physical
                      default         physical


        ADD                  DELETE                  EXIT

Use <Space> to select
```

To define a new extension select **ADD**.

```
V!CAS Setup Tool                                  BinTec Communications GmbH
[PABX] [EXTENSION] [ADD]: Configure PABX Extensions                    vicas

      Extension
      Type                          all
      User
      Destination                   application


      EAZ


            SAVE                                  CANCEL

Use <Space> to select
```

Extension = The number to which the following settings apply. If your V!CAS is connected to a point-to-point ISDN access the extension can be any number you like, if you have a point-to-multipoint configuration you will have to enter the final digit(s) of one of your MSNs (multiple subscriber numbers).

Note: The extension should only consist of digits (0-9). You should *not* use the special characters »#« and »*« as part of your extensions. Whether internal calls start with a »#« or not is defined in the PABX → STATIC SETTINGS menu.

Type = Specifies the type of calls this extension accepts.

Type	Accept calls for...
all	voice and data
voice	voice (telephone, fax, etc.)
data	data (applications)

User = The user who owns this extension.
In general each extension is assigned to one user.

Destination = The type of destination calls to this extension are connected to. There are four possible values:

Destination	Meaning
physical	A device connected to one of the POTS ports.
application	A TAPI or CAPI software application on your PC.
ppp	V!CAS's internal multiprotocol router.
isdnlogin	The isdnlogin facility of the system.

If *Destination* is set to **physical**, the POTS port selected under *Module* can be reached under this number from the other

POTS port for internal (i.e. toll-free) calls.
Default value: application

Depending on the type of destination you selected one or two of the following fields will also be visible:

Layer 1 Protocol = The layer1 protocol to be used for multi-protocol-routing (incoming calls only). (Visible if Destination = ppp)
Possible values:

Value	Meaning
auto	Default value, good for all connection types listed below (except for the specific PPP Modem Profile 2 ... 8 settings) if the calls are signalled correctly (as is the case in most of Europe). ***If in doubt, try this value.***
sync 64k	64kbps data connection
sync 56k	56kbps data connection
Modem	V!CAS: Selects Modem Profile 1 as configured in the [*MODEM*] menu
V.110 (1200 - 38400)	bit-rate adaption according to V.110 (1200 bps, 2400 bps, ..., 38400 bps)
Modem Profile 1 ... 8	V!CAS: Selects Modem Profile 1 ... 8 as configured in the [*MODEM*] menu

Default value: auto

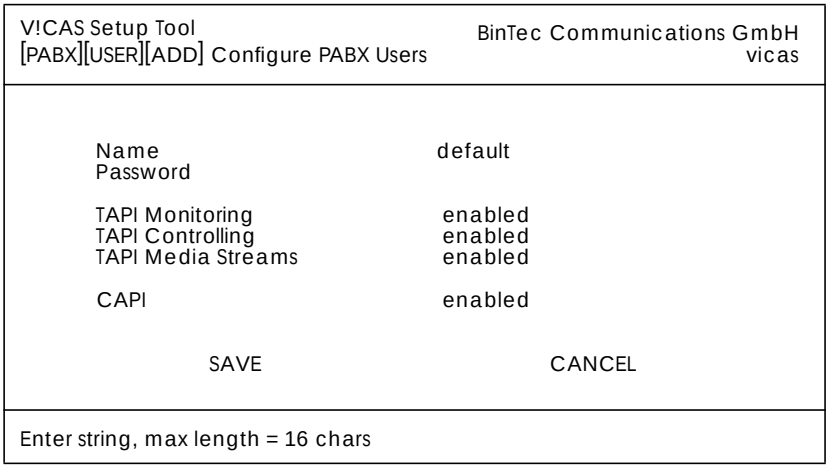
Interface = The interface name of the MPR interface (WAN Partner) to be used for the call. (Visible if Destination = ppp)
Default value: auto

Module = Phone 1 or Phone 2. (Visible if Destination = physical)

EAZ = The EAZ is only used by internally 1TR6-based applications such as CAPI 1.1. If you use a CAPI 1.1 application to access your V!CAS you have to enter a digit (0…9) here. (Visible if Destination = application)

PABX → USER →

This menu displays a list of all users currently configured. You can add new users, or change or delete existing ones. To configure a new user select ADD.



The users configured here can be selected in the User fields of the PABX → EXTENSIONS and CM-POTS, PHONE X menus. You will also have to enter the user name and corresponding password in the *BRICKware for BinGO! Plus/Professional* application (see online documentation on the Companion CD) if you want to access the POTS ports from TAPI or CAPI applications on your PC.

The User Concept

PABX Users were introduced to prevent anybody who has access to your PC—or the network your V!CAS is connected to—from using your V!CAS from CAPI or TAPI applications, which usually results in unwanted phone charges for your ISDN access.

You can therefore now define PABX Users on the V!CAS which serve exactly this one purpose—to restrict the access to your V!CAS's ISDN resources to authorized persons.

The system comes with one pre-defined user named **default**. This user is allowed to use all TAPI and CAPI features, and has no password.

If you only use the V!CAS from your PC or from a small network, where every network user shall be able to make use of the V!CAS's ISDN port, you do not need to add any new users, but can use the **default** user. To be able to access V!CAS from a PC application you will first have to configure a BRICK User in the *BRICKware for BinGO! Plus/Professional* application (see online documentation on http://www.bintec.de).

If you want to access V!CAS from CAPI 1.1 applications, you must use the **default** user as pre-defined in your system. Other users—even if CAPI is enabled—cannot access V!CAS from CAPI 1.1 applications.

You can configure the following parameters for each user:

Name = The name of the user.

Password = The password for this user.

TAPI Monitoring = Allow or deny the user to monitor call activity with TAPI applications.
Possible values: enabled, disabled
Default value: enabled

TAPI Controlling = Allow or deny the user to control calls for his extensions with TAPI.
Possible values: enabled, disabled
Default value: enabled

TAPI Media Streams = Allow or deny the usage of TAPI media streams.
Possible values: enabled, disabled
Default value: enabled
This parameter is not valid for V!CAS and its value is ignored.

CAPI = Allow or deny the usage of CAPI.
Possible values: enabled, disabled
Default value: enabled

In general each extension is assigned to one user.

MIB

A new set of PABX tables will be added to the MIB with this new Release 4.8.6. The new tables belong to the new PABX group:

```
g pabx
GROUP pabx (11):
 20 pabx 118 pabxUserTable 119 pabxTrunkTable
120 pabxTrunkPrefixTable 121 pabxExtensionTable
```

A description of the new tables you can find in the current MIB Reference at http://www.bintec.de/download/brick/doku/mibref/index.html

Which changes in configuration are necessary with this new tables is described in the section Changes on page 30.

At the same time two tables of the old MIB are deleted respectively ignored with this new Software Release. The table *potsIfTable* from the Interfaces group is deleted and the table *isdnDispatchTable* from the ISDN group stays empty.

ISDN Supplementary Services

Your V!CAS has a couple of built-in telephony services, which can be accessed from the telephones connected to the phone ports by dialling special codes starting with »*«.

At the moment the following functions are available:

** *dial a single * when necessary*

*0# *same as the Recall key* (often marked R or Hold)
If your telephone is equipped with a special R key, you can of course use this key instead of dialling *0#.

*1# *disconnect the current call*
Convenient when you have an active call and one call

on hold. Dialling ∗1# then terminates the active call and recalls the held call.

∗2# *enable call-waiting*
If you have an active call you will be made aware of a waiting call by a call-waiting tone.
After each power-up or reboot of your BinGO! Plus call waiting will be enabled automatically.

∗3# *disable call-waiting*
Additional incoming calls will be refused (*User busy*) when there is an active call. Calls already waiting will also be refused immediately.

∗4# *call forwarding …*

∗4∗0∗‹*No.*›#	… *always*
∗4∗1∗‹*No.*›#	… *when busy*
∗4∗2∗‹*No.*›#	… *when no answer*

where ‹*No.*› is the telephone number to forward the call to.
The codes ∗4∗0∗#, ∗4∗1∗#, and ∗4∗2∗# disable the corresponding call forwarding setting.

You may need to order the *call forwarding* feature from your telephone company before you can use it.
Also note that using *call forwarding* may incur additional charges.
Contact your telephone company for details.

∗5# *three-party conference*
When you have an active call and one call on hold, dialling ∗5# will connect you with both *external* calls in a three-party conference.

You may need to order the *three-party conference* feature from your telephone company before you can use it, and using *three-party conferences* may incur additional charges.
Contact your telephone company for details.

∗6# *terminate three-party conference*
When you have established a three-party conference

by dialling ∗5# you can terminate it again with ∗6#. This will return you to the state the calls were in prior to dialling ∗5#, i.e. one external call is on hold, the other external call is connected.
You can then switch between the two calls using the ∗0# combination (Hold).

∗7# *CLIR for the following call*
Prevents your ISDN number from being displayed on your partner's display for the next call only (CLIR is short for Calling Line Identification Restriction).

∗8# *Call Transfer*
When you have one call on hold (internal or external) by having dialed R or ∗0# and set up a second call (internal or external), you can connect these two calls by dialling *∗8#*. This function is available as soon as you hear the ringing for the second, active call.

If the second, active call is an external call the availability of this feature depends on your telephone company's supplementary service ECT (Explicit Call Transfer).

Microsoft Callback Extension to Mode 3

The Microsoft Callback Control Protocol (CBCP) knows different modes to decide which number is used for callback. This protocol is activated, when there is a call from a Windows95/ NT client.

Up to now Mode 2 was implemented. In Mode 2 (callback to a user-specifiable number) the user is asked, when calling from a Windows95/ NT client, to enter the callback number. This number is then used for callback.

From this release on the MS-CBCP was extended to Mode 3. Mode 3 uses a predefined number for callback.

Which mode is used (Mode 2 or Mode 3) depends on whether there is a predefined number assigned. When there is a predefined number, either a entry in the *biboPPPDialTable* for this partner (*Direction*: *both* or *outgoing*; *Type*: *isdn* or *isdn_spv*) or

when authentication is made via RADIUS and the RADIUS attribute Callback-Number is assigned, then Mode 3 is used. When calling from a Windows95/ NT client the caller is asked in a dialog box to confirm the mode (Mode 3) respectively the callback number. With no number assigned callback is made using Mode 2.

Such it is ensured that a callback is either made using the user-specified or the predefined number.

The variable *CallBack* in the *biboPPPTable* can be set to *ppp_offered* or *enabled*. But you must notice that with the value set to *enabled* no authentication is made during callback.

Also see Microsoft Callback via RADIUS below.

Microsoft Callback via RADIUS

With this new release it is possible to use Microsoft Callback via RADIUS for calls from a Windows95/ NT client.

The RADIUS server must be configured as follows:

Service-Type = Callback-Framed

Specifying only the Service-Type means using Mode 2 of the CBCP (user-specifiable number). This configuration assigns the value *enabled* to the variable *biboPPPCallBack* in the *biboPPPTable*.

To use Mode 3 (predefined number), that means using a fixed callback number, you must additionally assign a callback number as in the following example:

Service-Type = Callback-Framed

Callback-Number = “392”

The feature Microsoft Callback via RADIUS is only available for an inband identification of the caller (no calling line identification). The same it’s not possible to set the value *ppp_offered* for the variable *biboPPPCallBack*. For the time of the PPP connection there exists a temporary entry in the *biboPPPTable* with variable *CallBack* assigned the value *enabled*. This means that there is no additional authentication during callback. In Mode 3

a temporary entry in the *biboPPPDialTable* using the defined calling number is generated, too.

IPX RADIUS Extensions

The BRICK now supports dial-up IPX client connections via RADIUS. For a detailed description of this new feature see IPX RADIUS Extensions on page 41.

X.25

X.25 Window/ Packet Size Negotiation

Now you can decide for each X.25 link, whether a window / packet size negotiation is made.

x25LkPrNegotiation is the new parameter in the *x25LinkPresetTable*, which handles this feature. This parameter can be assigned three possible values:

never	No negotiation. When a call arrives that does not correspond to the default size, the call is cleared.
always	Negotiations are always made.
when_necessary	There are only negotiations, when the requested values differ from the default values.

Window / packet size negotiation settings can also be configured via Setup Tool, see "Configuring X.25 Parameters in Setup Tool" on page 21.

Configuring X.25 Parameters in Setup Tool

Now it is possible to configure additional X.25 parameters using Setup Tool.

• X.25 Link Configuration

X.25 → LINK CONFIGURATION → EDIT

Here window / packet size negotiation can be adjusted for an X.25 link. **Windowsize/ Packetsize Neg.** corresponds to the parameter *x25LkPrNegotiation* in the *x25LinkPresetTable*.

```
V!CAS Setup Tool                              BinTec Communications GmbH
[X.25][LINK][ADD]: X.25 Link Configuration                        vicas

        Link                                  en1-llc
        L3 Mode                               dte
        L3 Window Size                        default: 128      max: 128
        L3 Packet Size                        default: 2        max: 7
        Windowsize/Packetsize Neg.            when necessary

        Lowest Two-Way-Channel (LTC)  1
        Highest Two-Way-Channel (HTC) 2

        Partner MAC Address (LLC)

        Layer 2 Behaviour                     disconnect when idle

              SAVE                                  CANCEL

Use <Space> to select
```

Windowsize/ Packetsize Neg. = Decides whether window / packetsize negotiation is made for this X.25 link. The possible values are **never**, **always** and **when necessary**, where **when necessary** is the default value. The value *never* means no negotiation. When a call arrives that does not correspond to the default size, the call is cleared. *Always* means negotiations are always made and when *when necessary* is selected, there are only negotiations, when the requested values differ from the default values.

•WAN Partner

WAN PARTNER → ADD → ADVANCED SETTINGS

For WAN partners using the protocols X.25, X25ppp, X.31 B-Channel or X.25 no signalling the Layer 2 Mode can be configured in the advanced settings. The item Layer 2 Mode corresponds to the parameter *biboPPPLayer2Mode* in the *biboPPPTable*.

Layer 2 Mode = Layer 2 Mode can receive the values **auto**, **dte** or **dce**, where **auto** is the default value.

```
V!CAS Setup Tool                          BinTec Communications GmbH
[WAN][EDIT][ADVANCED]: Advanced Settings                        vicas

        Callback                          no
        Static Shorthold                  20  Idle for Dynamic Shorthold (%)0
        Delay after Connection Failure    300
        Dynamic Name Server Negotiationyes
        Channel-Bundling                  no



        Layer 1 Protocol                  ISDN 64 kbps
        Layer 2 Mode                      dte

              OK                                    CANCEL

Use <Space> to select
```

•WAN Partner Numbers Advanced

Here the item Closed User Group can be configured. The item corresponds to the parameter *biboDialClosedUserGroup* in the *biboDialTable*.

```
V!CAS plus Setup Tool                     BinTec Communications GmbH
[WAN][Extended]: Extended Settings of WAN-Partner Numbers       vicas

        Closed User Group                 none



              OK                                    CANCEL

Use <Space> to select
```

Closed User Group = The item Closed User Group can be assigned the values **none** or an integer from **1 to 9999**. **None** is the default value.

Active Layer 2 Set Up for Incoming X.25 Calls

Prior to release 4.8.6 the BRICK remained passive during setup of incoming X.25 dialup connections and waited for a SABM (Set Asynchronous Balance Mode) from the caller. Some X.25 implementations however were also waiting for a SABM from the BRICK.

Now the BRICK only waits one second for an incoming SABM. If no SABM is received within this time, the BRICK will send a SABM.

Because of the wait time the probability of a layer 2 setup collision is very small. Standard end-devices handle such collision correctly.

TP0 Bridge Extensions

To make possible connections between TCP clients and an X.25 network the TP0 Bridge feature (RFC 1086 / RFC 1006) is implemented on the BRICK.

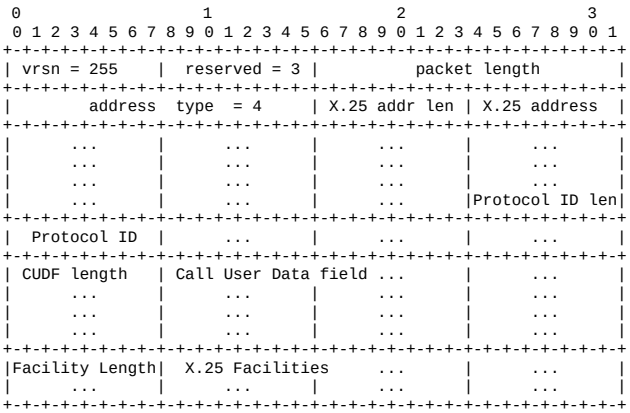
With this release two extensions concerning the transmission of X.25 data (RFC 1006) for incoming X.25 connections to the TP0 Bridge have been made.

Firstly with release 4.8.6 NSAP addresses, which are subaddresses of X.25 addresses, can be proofed for incoming X.25 calls. If a listener transfers a NSAP address in the facility field of the listening address, only X.25 calls with the same NSAP address are signaled to the listener.

The second extension concerns the X.25 call indication packet, which is sent as the first packet, when an incoming X.25 connection is established. Now with release 4.8.6 there is a possibility that the listening application gets some information about the contents of the X.25 call indication packet.

To get this data the value of the function byte, (the first byte, the listener sends to the TP0 brigde, see RFC 1086) has to be 66 instead of 2. Then the first data packet, the listener receives on

its new established TCP stream has the following format: It consists of 4 Byte TP0 header and the data in the extended X.25 Address Format:



IP Filter for TCP State and ICMP Type

The filters for IP access have been enhanced.

ICMP Type

The filters can now be used to filter IP packets in dependence of the ICMP type.

In the *ipFiltertable* there is the new variable *icmptype*, which can be assigned the following values :

echoRep, *destUnreach*, *srcQuench*, *redirect*, *echo*, *timeExcds*, *parmProb*, *timestamp*, *timestampRep*, *addrMask*, *addrMaskRep*, *dont_verify* .

Setup Tool's Filters menu has also been changed. You can now define filters according to appropriate ICMP types using the Type field after setting the protocol field to "ICMP".


```
V!CAS Setup Tool                          BinTec Communications GmbH
[IP][ACCESS][FILTER][ADD]: Configure IP Access Filter              vicas

    Description             echo request
    Index                   9

    Protocol                icmp
    Type                    echo

    Source Address
    Source Mask
    Source Port             any

    Destination Address
    Destination Mask
    Destination Port        any

                  SAVE                  CANCEL

Use <Space> to select
```

TCP Connection State

Filters can now be defined based on the state of an TCP Connection.

In the *ipFilterTable* there is the new variable *TcpConnState*, which can be assigned the following values:
dont_verify, established.

When this variable is set to *established*, this filter matches for TCP packets,which do not initiate a connection.

A typical application for this filter is to let packets pass through, which belong to connections that were initiated from inside, but discard all other TCP packets. This can be configured by the following rules:

1. rule: ALLOW (TCP / established)
2. rule: DENY (TCP / dont_verify)

The configuration in Setup Tool:

IP → ACCESS LISTS → FILTERS → ADD

```
V!CAS Setup Tool                              BinTec Communications GmbH
[IP][ACCESS][FILTER][ADD]: Configure IP Access Filter                vicas

    Description             TCP established
    Index                   10

    Protocol                tcp
    Connection State        established

    Source Address
    Source Mask
    Source Port             any

    Destination Address
    Destination Mask
    Destination Port        any

                  SAVE                  CANCEL

Use <Space> to select
```

New Trace Command Feature

The trace command has been enhanced. It is now possible to decode HOLD and RETRIEVE messages in the D-Channel.

Status Display for Modems

The web based status page for your BRICK now additionally displays information on installed modems.

Under Hardware Interfaces you will find the modems with the respective slot they are installed in. The modem type and

modem status are displayed. Each modem used at the moment is marked red and when you move the mouse pointer over the red channel symbol, the rate for receiving and transmitting data in bps is displayed. Additionally you are informed about the ISDN channel used. The four digits xyzz stand for the slot (x), the unit (y) and the ISDN channel used (zz). For the example below this would consequently mean: slot 2, unit 0 and channel 1.

Wildcards for Dialing Numbers

Similar to wildcards for the calling party's address in the *biboDialTable* for incoming calls, the variable *Number* now can also contain wildcards for outgoing calls. The wildcards for outoing and incoming calls are defined as follows:

Wildcard	Example	Outgoing Calls	Incoming Calls
*	1234*	is ignored, e.g 1234	matches zero or any string, e.g 1234 or 123467
?	1234?	is replaced by 0, e.g. 12340	matches any single digit, e.g. 12349, 12347
[a-b]	123[5-9]	first digit in the range, e.g. 1235	denotes the range of possible digits to match, e.g. 1235, 1236
[^a-b]	123[^0-5]	range of digits not allowed, first possible digit inserted, e.g. 1236	denotes the range of excluded digits to match, e.g. 1236, 1237
{ab}	{00}1234	inserted for outgoing calls, e.g. 001234	optional string to match, e.g. 001234, 1234

The advantage is, that now you can use one entry for the variable *Number* for incoming and outgoing calls. For Example {0}91196790 will generate 091196790 for outgoing calls and will accept 091196790 and 91196790 for incoming calls as valid CLID.

Link Quality Monitoring

By the help of Link Quality Monitoring (LQM defined in RFC 1989) it is possible to exchange information within a PPP connection to draw conclusions about the underlying connection quality.

This information is typically transmitted periodically to the partner as so-called Link Quality Reports (LQR). The interval (Reporting Period) is agreed upon during the LCP negotiation.

Link Quality Monitoring can be useful to examine e.g. modem connections. (With unreliable modem connections it can happen that because of CRC errors no more data can be transmitted.)

For detailed information on the new feature Link Quality Monitoring see the section Detailed Feature Descriptions on page 44.

Extended Syslog Messages

The syslog messages have been extended for a detailed analysis of fax connections.

Changes

D-Channel Protocol

The new PABX concept, which is included in release 4.8.6, only supports the DSS1 (point-to-point and point-to-multipoint) as D-channel protocol (Euro-ISDN).The same leased lines are no longer supported.

Priority Voice Technology

Priority Voice technology, what is a feature of V!CAS, is not included in the features of software release 4.8.6.

This means that with this release the functionality of using both B-channels for data transfer and at the same time making or receiving phone calls is lost.

As soon as possible the Priority Voice feature will be included into the software again.

Changes in Setup Tool Configuration

In the following you find a description of the differences in configuration of V!CAS, which are necessary after you have installed the Software Release 4.8.6.

1 WAN Interface

In this menu now additionally Country Code, Area Code and Subscriber Number of the BRICK must be configured

This menu contains settings for the ISDN interface.

```
V!CAS Setup Tool                          BinTec Communications GmbH
[WAN]: WAN Interface                                          vicas

Result of autoconfiguration:    Euro ISDN, point to point

ISDN Switch Type                autodetect on bootup

Country Code                    44
Area Code                       115
Subscriber Number               1234

Advanced Settings>

                SAVE                          CANCEL

Use <Space> to select
```

Result of autoconfiguration = The status of ISDN autoconfiguration for this interface. The auto detection procedure runs until a successful detection or the switch type (see below) is set manually.

ISDN Switch Type = Defines the switch type your ISDN provider uses. In most cases “autodetect on bootup” will detect the proper switch type. If the switch type is set manually, the auto detection feature is disabled for this interface.

The Euro ISDN (DSS1) protocol is supported for dialup lines.

The following three entries Country Code, Area Code and Subscriber Number are mandatory, if your V!CAS is connected to an ISDN point-to-point access (called *Anlagenanschluß* in Germany). If connected to an ISDN point-to-multipoint access the entries are optional, but if you wish to fill them in, take care to enter the correct values, otherwise you will not be able to establish connections to and from your BRICK.
For example, if you have an ISDN point-to-multipoint access with three MSNs, 2345, 2346, and 2347, and you wish to use the

last digit of these MSNs as the extension number for devices connected to the POTS ports, you would have to enter **234** in the *Subscriber Number* field.

Country Code = The international dial prefix for the country your V!CAS is located, e.g. 49 for Germany, or 44 for the UK.

Area Code = The dial prefix for the area (or city) your V!CAS is located, e.g. 911 for Nürnberg, Germany, or 115 for Nottingham, UK.

Subscriber Number = The number of your ISDN access.
These three entries are mandatory, if your V!CAS is connected to an ISDN point-to-point access (called *Anlagenanschluß* in Germany). If connected to an ISDN point-to-multipoint access the entries are optional, but if you wish to fill them in, take care to enter the correct values, otherwise you will not be able to establish connections to and from your BRICK.
For example, if you have an ISDN point-to-multipoint access with three MSNs, 2345, 2346, and 2347, and you wish to use the last digit of these MSNs as the extension number for devices connected to the POTS ports, you would have to enter **234** in the *Subscriber Number* field.

2 Call Answering

The call answering is no longer configured in the menu [WAN] under

but in the menu

which is described in detail in the paragraph Setup Tool under Features.

If you have received more than one MSN with your ISDN access you can use the different numbers to route calls to the appropriate destination.

Let's assume you received the three MSNs 7654, 7655 and 7656 with your ISDN point-to-multipoint access, and you want to connect analog telephones to Phone ports 1and 2.

The telephone at port 1 should accept all voice calls for MSN 7654, the telephone at port 2 should accept voice calls for MSN 7655. In addition to that data calls for these two MSNs should be routed to the *isdnlogin* service, while all calls for MSN 7656 should be routed to the *ppp* service.

Before you begin

You need to know your ISDN telephone numbers, which of these numbers to use for which destination, and which device is connected to which Phone port.

Configure it

PABX → EXTENSIONS → ADD **Add Extension**

Start with extension **4** (the last digit of your MSN 7654) and select **voice** in the *Type* field. Only set a *User* if you want to control this extension from a PC. Select **physical** in the *Destination* field, and **Phone 1** in the *Module* field. (The *Module* field will appear as soon as you leave the *Destination* field (if set to **physical**).) Save this extension.

Then add and save the other extensions. The following table lists the required settings for all extensions:

Extension	Type	Destination	Module
4	voice	physical	Phone 1
5	voice	physical	Phone 2
4	data	isdnlogin	–
5	data	isdnlogin	–
6	all	ppp	–

For CAPI-Listener applications (e.g. faxserver applications) an entry with the value **application** under the item **destination** must be generated. The MSN resp. extension used must also be configured in the CAPI application. When a user is defined here, this user also has to be defined in the menu [PABX][USER] with CAPI=enabled and in the configuration of the CAPI/TAPI drivers on the PC (contained in BRICKware for BinGO! Plus/Professional/V!CAS).

Changes in Configuration Via SNMP

The changes in configuring the BRICK affect the configuration via SNMP, as well as the Setup Tool. The new PABX tables are described under MIB respectively in the MIB Reference on BinTecs Webserver at http://www.bintec.de/download/brick/doku/mibref/index.html

1 D-Channel Protocol

The ISDN stack resp. the D-channel protocol is no longer configured or read out in the *isdnStkTable*, but in the *pabxTrunkTable*.

New configuration:

```
vicas:pabxTrunkTable> pabxTrunkTable

inx number(*rw)      Descr(rw)          Slot(rw)          Unit(rw)
    Protocol(-rw)    Config(rw)         TeiProc(rw)       TeiValue(rw)
    CountryCode(rw)  AreaCode(rw)       SubscriberNo(rw)  Extension(rw)

00  0                                   2                 0
    dss1             point_to_multipoi  automatic         0
```

Old Configuration:

```
vicas:isdnStkTable> isdnStkTable

inx Number(*rw)      IsdnIfIndex(rw)    ProtocolProfile(-rw)
    Configuration(rw) SPID(rw)          TeiProc(rw)
    TeiValue(rw)     ClearAllCalls(rw)  Status(ro)
```

```
    Layer2State(ro)        Bchannels(rw)        DialOutPrefix(rw)
```

Where the new *Protocol* corresponds to the old *ProtocolProfile* and the new *Config* corresponds to the old *Configuration*.

2 Call Answering

Call answering is no longer configured in the isdnDispatchTable but in the pabxExtensionTable.

New configuration:

```
vicas:pabxExtensionTable>

inx Extension(*rw)      Type(rw)          Destination(-rw)      Slot(rw)
    Unit(rw)            IpAddr(rw)        User(rw)              EAZ(rw)
    Layer1Prot(rw)      IfIndex(rw)

00 "4"                  voice_and_data    isdn_login            0
    0           0.0.0.0
    auto        0

01 "3"          voice               physical                  3
    1           0.0.0.0             "user2"
    auto        0

02 "2"          voice               physical                  3
    0           0.0.0.0             "user1"
    auto        0

03 "1"          data                multiprotocol_rou         0
    0           0.0.0.0
    auto        0

04 "1"          voice               application               0
    0           0.0.0.0             "faxuser"                 "1"
    auto        0
```

Old Configuration:

```
vicas:isdnDispatchTable>

inx StkNumber(*rw)          Item(*-rw)          LocalNumber(rw)
    LocalSubaddress(rw)     Bearer(rw)          Slot(rw)
    Unit(rw)                Direction(rw)       Mode(rw)
```

In the new configuration with the PABX concept *Slot* and *Unit* are only significant for physical interfaces.

A new feature is, that to every extension you can assign a user (see The User Concept).

The *isdnDispatchTable* still exists in the PABX products, but stays empty at present.

The new *Extension* corresponds to the old *LocalNumber*, *Type* to *Bearer* and *Destination* to *Item*.

Bugfixes

The bugfixes noted here concern the router part of V!CAS software, which still is a part of the new PABX-based software. These are bugfixes since Software Release 4.8 Revision 3.

Known Bug

- Dial-up connections for RADIUS-Users
  Interfaces configured to use `ip_lapb` encapsulation, using the following entry in `/etc/raddb/users`,
  `BinTec-biboPPPTable = "Encapsulation=ip_lapb"`
  are sometimes rejected by the BRICK.

TP0 Bridge

- Sometimes it happened that it was not possible to establish an initial TP0 Bridge connection (RFC 1086) between the TCP client and the BRICK.

  This bug has been fixed. In this context the TP0 Bridge syslog messages have been changed, too.

X.31 in D-Channel

- When CAPI applications were using X.31 in D-Channel for X.25 packet switching, the variable *AssignedTo* in the *isdnDChanX31Table* was not considered (Setup Tool: Advanced settings for the WAN interface).

  This bug has been fixed.

  The variable *AssignedTo* can be assigned the following values:

packet_switch	The TEI may be used only by the X.25 router.
capi	The TEI may only be used by a CAPI application and this TEI has to be configured also in the CAPI application.
capi_default	The TEI may be used only by a CAPI application and the BRICK overwrites the

	TEI value that is configured in the application.
delete	Sets the table entry to delete.

CAPI

- If a CAPI application used an X.25 protocol, it wasn't any longer informed about incoming X.25 calls after it has sent a CONNECT_B3_REQ message to establish an outgoing X.25 connection. In this case the CONNECT_B3_Ind message, the application has to receive got lost.

  This bug has been fixed.

Dynamic Shorthold

- When combining dynamic B-Channel Bundling with optimally making use of the charging intervals (by dynamic shorthold), there was the problem that, when reducing the bandwith, the current charging intervals were not taken into consideration. This bug has been fixed.
  To optimize charges now the bandwith is reduced by disconnecting a B-Channel only short before a new charging interval.

STAC Compression on Multilink PPP Interfaces

- According to RFC 1974 (*PPP STAC LZS Compression Protocol*) there are several check modes to keep the compressor and decompressor histories in synchronisation even in the absence of a reliable link to guarantee the sequential transmission of data.

  In rare cases it still happened that, when *biboPPPCompression* = stac (RFC 1974, check mode 3) was used on MultiLink PPP interfaces the history re-synchronisation process sometimes came to a state where decompression histories were out-of-sync and user data could no longer be transmitted over the line.

  This bug has been fixed in the current release.

Spaces in biboPPPLoginString

- There appeared problems in the login procedure configuration (especially for Compuserve users), when strings like passwords or login names were containing spaces. That was because spaces are used as internal flags to handle the login procedure.

  To handle this problem blank spaces in strings, which are part of the variable *LoginString* in the *biboPPPTable*, must be preceded by a backslash as shown in the following example for the string “pass word”:

```
inx LoginString(rw)
00 "-d1 \n e: CIS\n D: name/go:pppconnect\n wor -d1 pass\ word\n PPP"
```

  This must be considered, when the variable *LoginString* is configured via SNMP. In the Setup Tool no additional backslashes have to be entered, when configuring the items Host, User ID and Password in the menu [WAN][EDIT][ADVANCED][PROVIDER].

Setting Administration Status to Down

- When by “ifconfig down” or via the Setup Tool the variable *ifAdminStatus* was set to down for an active interface, the variables in the PPP accounting syslog message containing PPP connection information only had the value 0.

  This bug has been fixed. Now the syslog message contains the correct values.

Setup Tool

- When a PPP interface was resetted in the interface monitor in [MONITOR][INTERFACES][EXTENDED], there could occur a reboot of the BRICK with large configurations.

  This bug has been fixed.

- When a default route was configured for a WAN partner interface in the [IP][ROUTING] menu or the SNMP shell and afterwards the [WAN][EDIT][IP] menu was opened

again for this WAN Partner and left with SAVE, this default route was deleted.

This bug has been fixed.

Modem

- When a modem connection was active on the V!CAS a second data connection sometimes attempted to access the same modem and resulted in a system reboot. This problem has been corrected.

Detailed Feature Descriptions

IPX RADIUS Extensions

The BRICK now supports dial-up IPX client connections via RADIUS. Support for IPX links via RADIUS has been tested using Merit's AAA RADIUS server 3.5.6. The examples shown below can be used with the Merit server, for use with other servers consult your local documentation.

Assuming a RADIUS server has been configured in Setup Tool's IP → RADIUS SERVER menu (or the *radiusServerTable* from the SNMP shell), and the RADIUS server can successfully authenticate the caller, dial-up links can be setup to support IPX networking using standard IPX or BinTec-specific RADIUS attributes mentioned below.

Standard Attributes

Framed-IPX-Network

This attribute defines a transfer network for the IPX link. Setting this attribute to "Framed-IPX-Network = 8" effectively sets the IPX network number for the transfer network to "0:0:0:8".

Normally, when this attribute is set to "ffffffe" the calling host is assigned a network number from an existing address pool, currently this feature is not supported on the BRICK. To disable a transfer network for the IPX WAN link you can set this attribute to "0" (no transfer network, unnumbered RIP).

RIP / SAP updates: If the "Framed-IPX-Network" attribute is used entries in the BRICK's *ripCircTable* and *sapCircTable* are configured using default settings for RIP / SAP updates (triggered+piggybacked).

BinTec-specific Attributes

BinTec-specific RADIUS attributes added in release 4.8.3 are defined below. For additional information, see also the section *Using BinTec-specific Attributes* on page 43.

BinTec-ipxCircTable	Creates or modifies ***ipxCircTable*** entries.

BinTec-ripCircTable	Creates or modifies ***ripCircTable*** entries
BinTec-sapCircTable	Creates ore modifies ***sapCircTable*** entries.

Example IPX over RADIUS /etc/raddb/client Entries

The definitions below could be used for a dial-up IPX link between two BRICKs. RIP / SAP updates will be performed via the default triggered+piggybacked mode.

```
kornburgxl
    Password = "access2roth", Framed-IPX-Network = 0
    Framed-Protocol = PPP, Idle-Timeout = 300,

rothxl
    Password = "access2kornburg", Framed-IPX-Network = 0
    Framed-Protocol = PPP, Idle-Timeout = 300,
```

If the router rothxl supports IPX WAN links but requires a transfer network (0:0:0:9) this definition could be used instead.

```
rothxl
    Password = "access2kornburg", Framed-IPX-Network = 9
    Framed-Protocol = PPP, Idle-Timeout = 300,
```

If the required IPX services are statically configured, RIP and SAP can be disabled for the WAN link using the BinTec-specific options shown below.

```
kornburgxl
    Password = "access2roth", Framed-IPX-Network = 0
    Framed-Protocol = PPP, Idle-Timeout = 300,
    BinTec-ripCircTable = "ripcircstate=off",
    BinTec-sapCircTable = "sapcircstate=off"
```

The definition below could be used for a dial-in Windows Client that does not support IPX WAN links. In this example, setting "Update = 0" disables periodic updates for active links.

```
winlaptop
    Password = "486pentium?"
    Framed-Protocol = PPP, Idle-Timeout = 300,
    Bintec-ipxCircTable = "netnumber=0:0:0:a ipxcirctype=ipxcpWS"
    BinTec-ripCircTable = "Update=0 AgeMultiplier=10000",
    BinTec-sapCircTable = "Update=0 AgeMultiplier=10000"
```

RIP/SAP Updates for RADIUS interfaces

Since RADIUS interfaces are only available as long as the re-

spective client is connected please note the following effects this may have on links configured for active RIP and SAP updates.

- Access to services on the dial-in client's LAN (from hosts on the RADIUS client's LAN) may not be reliable.
- IPX clients cannot be informed of changes (routing or service advertisements) unless they are actually connected when the change occurs.

This may lead to a state where a server appears as being present on the remote network but is no longer available. The preferred solution, albeit time-consuming, to this problem is to statically configure the required routes and services on the clients and to disable RIP/SAP updates.

Using BinTec-specific Attributes

Each of the BinTec-specific RADIUS attributes corresponds to a MIB table. Supported BinTec-specific attributes can be used in your server's /etc/raddb/users file. The attribute definitions must also be added to your dictionary file (normally found in /etc/raddb). To modify a MIB table entry you must use the following syntax:

<*BinTec-Option*> = *"variable*1=*value*1 … *variable*n=*value*n*"*

An example authentication line from a RADIUS /etc/raddb/users file might look like this:

```
Service-Type = Framed,
BinTec-biboPPPTable = "DynShorthold=50 IpAddress=static",
BinTec-ipNatPresetTable = "Protocol=tcp extport=1050 intport=100"
```

Also, when using these attributes please note:

- The table entry's *ifIndex* is set automatically and can't be influenced.
- The entries are not case-sensitive.
- You must not use blank spaces before or after »=« signs inside the double quotes.
- Attributes support either **static** or **dynamic** mode.

Static mode modifies existing table entries while dynamic mode creates a new table entry. All variables you want to create (dynamic) must be defined in one line.

Link Quality Monitoring

Because Link Quality Monitoring as described under Features is specified within LCP negotiation, i.e. before the authentication of the partner, for the configuration of incoming calls a distinction must be made between inband and outband identification.

In case of outband identification (CLID / outband RADIUS) and for outgoing calls the LQM is activated by setting the variable *biboPPPLQMonitoring* in the *biboPPPTable* to on. When a RADIUS server is used the variable is set by the help of the BinTec dictionary.

For incoming calls identified inband (identification by the internal *biboPPPTable* or via RADIUS server) the variable *biboPPPProfileLQMonitoring* in the *biboPPPProfileTable* must be set to on.

After a successful LCP negotiation for every link of a temporary connection additionally to the entry in the *biboPPPLinkTable* a correlating entry in the *biboPPPLQMTable* is generated. Both entries can be uniquely assigned to each other by the *IFIndex* respectively the *CallReference* value.

The *biboPPPLQMTable* is a new table and is described in detail in the following.
biboPPPLQMTable:

inx	IfIndex(*ro	CallReference(ro)	ReportingPeriod(ro)
	OutLQRs(ro)	OutPackets(ro)	OutOctets(ro)
	InLQRs(ro)	InPackets(ro)	InOctets(ro)
	InDiscards(ro)	InErrors(ro	PeerOutLQRs(ro)
	PeerOutPackets(ro)	PeerOutOctets(ro)	PeerInLQRs(ro)
	PeerInPackets(ro)	PeerInOctets(ro	PeerInDiscards(ro)
	PeerInErrors(ro	LossedOutLQRs(ro)	LossedOutPackets(ro)
	LossedOutOctets(ro)	LossedPeerOutLQRs(ro)	LossedPeerOutPkts(ro)
	LossedPeerOutOcts(ro)		

The *biboPPPLQMTable* contains statistical information for each current PPP link on the system. Only the system can add or delete entries to this table.

Entries are created by the system each time a new PPP link was established and LQM was negotiated successfully.

Entries are removed by the system, when the corresponding PPP link is disconnected.

For detailed information on the meaning of the single variables see the MIB Reference on the BinTec Website at http://www.bintec.de.