

RELEASE NOTE

V!CAS
February 22, 1999

New System Software: *Release 4.9 Revision 4*

Important Info: This document describes the new features, enhancements, bug-fixes, and changes to the V!CAS System Software for Release 4.9 Revision 4.



We recommend having a look at BinTec's WWW server at <http://www.bintec.de> (Section: FTP Server), where you can always find current, up-to-the-minute information about products and software releases.

Together with this Software Release 4.9.4, a new BRICKware is available, which can be found on the current BinTec ISDN Companion CD or retrieved from BinTec's WWW server (Section: FTP Server). The features Call Pickup (page 24) and Priority Voice (page 25) are contained in this Software Release.

Please note that with Release 4.8 Revision 6 a new software concept was introduced, which is described under [The Voice Data Product Line](#) on page 9. For V!CAS, Software Release 4.8 Revision 6 meant a functional shift from a router with a/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, do not update to Release 4.8 Revision 6 or Release 4.9 Revision 4 and leave Release 4.8 Revision 3 as the current software version of your V!CAS.

If you do update from a software version older than 4.8.6 to either 4.8.6 or 4.9.4, because of the new software concept, a spe-

cial update procedure is given that must be followed (see [Upgrading System Software II](#) on page 6).

The new BRICKware must also be installed if you are upgrading from a Software Release older than Release 4.8 Revision 6 to support the PABX functionalities. BRICKware for Windows (current version 4.9.4) can be retrieved from BinTec's FTP Server at <http://www.bintec.de> and should be installed together with Software Release 4.9 Revision 4 for V!CAS.

In case you are updating from software version 4.8.6 to 4.9.4, you can conduct the default update procedure, as described under [Upgrading System Software I](#) on page 5.

Please, also take notice of the [New Update Procedure](#) described on page 13, which concerns the default update procedure, for example, from Software Release 4.8.6 to 4.9.4.



With the PABX functionality, a user concept was introduced beginning with Software Release 4.8 Revision 6, which is described under [The PABX User Concept](#) on page 57.

All Changes that result from this new software concept that started with Release 4.8.6 are described in the section [Changes](#) on page 78.

Significant changes have been made concerning the CAPI and TAPI port numbers described under [New CAPI and TAPI Ports](#) on page 16 and [Proxy ARP](#) described on page 19. Please read the descriptions carefully, because changes in configuration are required for these features.

Contents:

Upgrading System Software II	6
The Voice Data Product Line	9
What's New in Release 4.9.4	11
Features	11
New BRICKware available	11
Setup Tool Menu Reorganization	11
Partner-Specific/non-Specific PPP Settings	13
New Update Procedure	13
Transferring Configuration Files via the Serial Port	14
New CAPI and TAPI Ports	16
IP Route Announcement	19
Proxy ARP	19
Access Lists	21
X.25 Dialout Without Configuration	21
Pools for Dynamic IP Address Assignment	22
WINS (NBNS) Negotiation over PPP	23
DHCP Server Functionality	23
X.25 in Setup Tool	23
PABX: Call Pickup	24
Priority Voice	25
Credits Based Accounting System	26
Bridging	27
New Timer in x25LinkPresetTable	28
Changes	29
TCP Optimization	29
Configuration: State File	29
CAPI: PLCI and NCCI	29
Charging Information	29
CAPI Syslog Messages	30
CAPI DATA_B3_IND message	31
PABX: Idle Tone	31
PABX: Configuration during an Established Call	31
Bugfixes	32
isdnLoginOnPPPDispatch	32
LAPB Encapsulation with Compression	32
biboPPPLQMTTable	32
CAPI	33
Network Address Translation	34
RAS and Remote TAPI (BRICKware)	34
localUdpAllowTable	34
Call Collisions with MS Callback	35
ifconfig Command	35



Setup Tool: WAN Partner	35
Setup Tool: Access Lists	35
Reboot when Establishing ISDN Connections	36
X.25 Routing Priorities.	36
NAT on a Dial-Up Interface.	36
Setup Tool: WAN Partner Configuration	37
Fax: T30 Carrier detection (V.21)	37
Accepting Calls with CAPI 1.1 Applications.	37
IPX: ripCircTable and sapCircTable	38
IPX: Configuring the NetNumber	38
HTTP Server: Internet Explorer 4.0	38
ISDN S ₀ : Auto Configuration	39
Bridging.	39
Fax Applications with Protocol Switching.	39
CAPI and Incorrect Bearer Capability	40
TAPI: Calls Sent to TAPI Clients Several Times	40
PPP Callback Working on the 2nd Attempt	40
Problems Accessing Compuserve for the First Time	40
Known Issues	42
Connection Attempts between Client and Server	42
Loading of an old VICAS Configuration via TFTP.	42
Outgoing FTP Connections via NAT.	42
Autologout Interrupting the Update.	43
Several Extensions for one POTS Port	43
Detailed Feature Descriptions	44
New Setup Tool PPP Configuration Options	44
IP Address Pools	46
Credits Based Accounting System	50
What Was New in Release 4.8.6	56



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 I

This upgrading procedure describes updating your V!CAS Software from Software Release 4.8.6 to Software Release 4.9.4.

1. Retrieve the current system software image from BinTec's WWW server at <http://www.bintec.de> (Section: FTP Server).
2. With this image 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. Information on using the BOOTmonitor can be found in the V!CAS *User's Guide* under *Firmware Upgrades*.
3. Please note that there is a new update procedure in case there is not enough memory available to perform a software update via the **update** command from the SNMP shell. The new incremental update loads the new software image in blocks of 64 KB via TFTP and writes it to the flash ROM immediately. Because this procedure offers no possibility to check the integrity of the image, please first use the option **"-v"** that verifies the image file. For more detailed information see [New Update Procedure](#) on page 13.
4. Once you've installed Release 4.9 Revision 4 you may want to retrieve the latest documentation (in Adobe's PDF format), which is also available from BinTec's file server at the address noted above.

Note: When upgrading system software, it is also recommended that you use the most current versions of *BRICKware for Windows* and *UNIXTools*. Both can be retrieved from BinTec's file server.

BRICK-XS-2MB systems ship with a new firmware logic version which has been made available for all BRICK-XS (BRICK-XS-1MB, BRICK-XS Office/1 and 2MB), BinGO!, and V!CAS systems. To determine which logic file you need for your product and how to perform the update, please refer to the [RN LOGIC.pdf](#).

Upgrading System Software II

This upgrading procedure describes updating your V!CAS Software from an Release version older than 4.8.6, to Software Release 4.8.6 or 4.9.4.



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 (4.9.4).

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 (4.9.4) 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 80.
6. Once you've installed Release 4.8 Revision 6 (Release 4.9 Revision 4) 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 (4.9.4), it is absolutely necessary that you use the most current version of *BRICKware for Windows* (Rel. 4.9 Rev. 4). It can be retrieved from BinTec's FTP server.

BRICK-XS-2MB systems ship with a new firmware logic version which has been made available for all BRICK-XS (BRICK-XS-1MB, BRICK-XS Office/1 and 2MB), BinGO!, and V!CAS systems. To determine which logic file you need for your product and how to perform the update, please refer to the [RN LOGIC.pdf](#).

Also pay attention to the following information, which is only relevant, when you are updating to Software Release 4.8.6. Software Release 4.9.4 has solved this problem by introducing a

new option, described under [New Update Procedure](#) on page 13.



Performing a software update on a running system (via the **update** command) currently requires that a contiguous block of free memory, \geq 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.

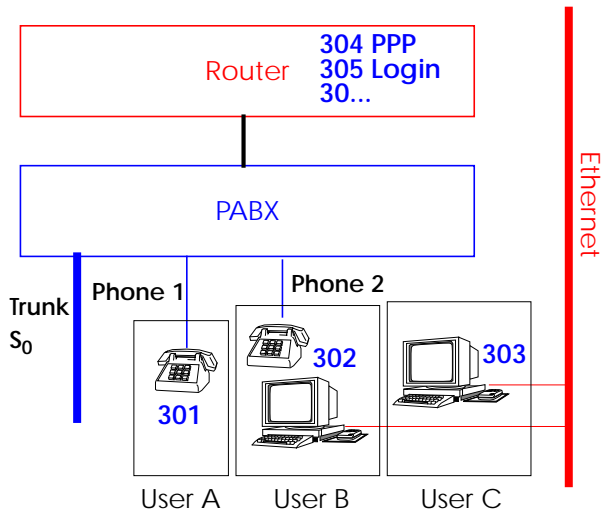
Note that you can always perform an update using the BOOTmonitor. The internal procedure of performing software upgrades on the BRICK is currently being optimized and a change is planned for a future release.

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 4.9.4

Release 4.9 Revision 4:

Features:

Bugfixes:

Detailed Description:

Features

New BRICKware available

There is a new BRICKware version 4.9.4 available. BRICKware 4.9.4 is contained on the current BinTec ISDN Companion CD or can be retrieved from BinTec's WWW server (Section: FTP-Server) at <http://www.bintec.de>.

Setup Tool Menu Reorganization

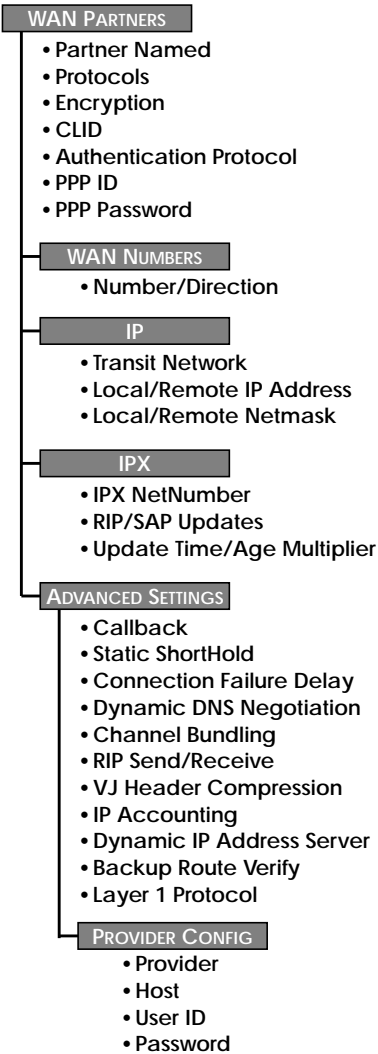
The Setup Tool's Partner Management menus (contained in the **WAN Partners** section) have been reorganized to reflect a more logical structure. With the exception of several new configuration options ([described below](#)), these changes only affect the menu structure. Protocol-specific configuration settings in Setup Tool's **WAN Partners** menu have been moved to the new/updated **PPP**, **IP**, **IPX**, and **BRIDGE** submenus.

As a guide to the new menu structure, the diagram shown on the following page shows both the old and the new menu layouts.

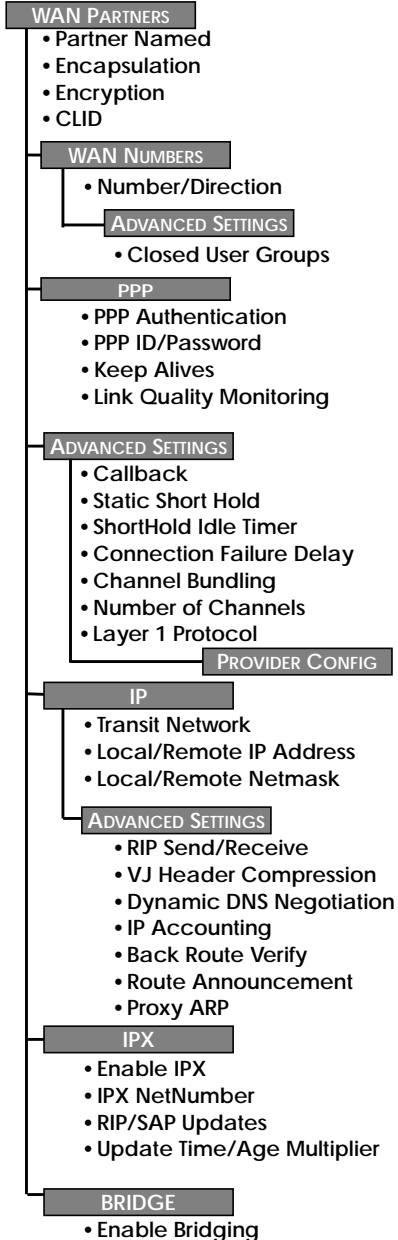
Also note that the new menu structure outlined below is also reflected in Setup Tool's **X.25** → **MPR** → , **FR** → **MPR** → , and **VPN** → submenus.

Detailed information about new configuration options appearing in this release can be found in the section [Detailed Feature Descriptions](#) on page 44.

Old Setup Tool WAN Partner Menus



New Setup Tool WAN Partner Menus



Partner-Specific/non-Specific PPP Settings

In Setup Tool partner-specific PPP settings can be configured under the new **WAN Partners** → **PPP** → menu. For information regarding the available partner-specific settings, see [New Partner-Specific PPP Settings](#) on page 45.

Default PPP settings (partner non-specific) can also be configured via the new **PPP** → menu. For information regarding these settings refer to [Global PPP Settings Menu](#) on page 44.

New Update Procedure

Performing a software update on a running system via the **update** command (SNMP shell) requires that a contiguous block of free memory, greater than or equal to the size of the new software image, is available. In the past there occurred problems for BRICKs with 4 MB RAM when the update application could not allocate enough memory to load a software image into RAM via TFTP.

With Release 4.9.4, the Update procedure was enhanced so that also in this case an update via the **update** command is possible.

When there is not enough memory available to load the complete image into RAM, the user is offered an incremental update. Then the new software image is loaded in blocks of 64 KB via TFTP and written to Flash ROM immediately. Because this procedure offers no possibility to check the integrity of the image, there is the option **-v** that verifies the image file.

Note that regardless of software image size and available RAM, you can always perform an update using the BOOTmonitor.

The following is an example of the new update procedure (verifying the image file and updating the BRICK):

1. Verifying the image

```
vicas:> update -v tftpserver vic494.vc
Starting File Transfer..... OK (754)
Checking new image... OK
File verifies OK
```

2. Updating the V!CAS with the image

```
vicas:> update tftpserver vic494.vc
Starting File Transfer .
Your current software release is 4.8.6.
New image has release 4.9.4.
```

```
WARNING: There is not enough free memory (RAM) to store
the new software image before writing it to flash. You
can perform an incremental update (the image is written
directly to flash in 64 KB increments). If you
need to perform an incremental update you should restart
the update using the -v option to verify the integrity
of the new file.
```

```
Don't reboot the router during the update.
```

```
Do you want to perform an incremental update (y or n)
[n] ? y
Receiving and Writing to FlashROM .....
Software update complete
Reboot now (y or n) [n] ? y
```



We recommend firstly verifying the software image file and then starting the incremental update after a successful verification of the file.

Transferring Configuration Files via the Serial Port

With Software Release 4.9.4, it is possible to load and save configuration files via the serial interface using the protocol XMODEM (up to now only possible via TFTP). Therefore, the variable *file* is assigned the value **xmodem** or **xmodem-1k**. **xmodem-1k** uses a packet size of 1024 Byte (default: 128 Byte) and in general reaches a higher throughput. The packet size is defined by the sender so that the value **xmodem-1k** only makes sense on the sending end; on the receiving end it is ignored.

To make use of this new feature you have to access your V!CAS from a computer via the serial port and a terminal program as described in “Getting Started” in the Chapter “Configuration” (“Over Serial Port”).

Getting the Configuration

```
cmd=get file=xmodem path=new_config
```

loads a file received via XMODEM with the name `new_config` into the flash ROM of the V!CAS.

After this command has been started, the terminal program must be set to Send (Upload) and the transmission protocol (XMODEM) as well as the source file name and location must be entered. The console cannot be used for the duration of the file transfer.

Putting the Configuration

```
cmd=put file=xmodem path=boot
```

sends the V!CAS' flash ROM file `boot` via XMODEM.

After this command has been started, the terminal program must be set to Receive (Download) and the transmission protocol (XMODEM) as well as the destination file name and location must be entered. For the time of the file transfer The console cannot be used for the duration of the file transfer.

Transmitting State Information

The previously mentioned commands only send or retrieve the configuration files containing variables with read/write status. They send/retrieve information from files stored in the flash. Using “`cmd=state`”, you can save all configuration information currently in the memory. This information includes read/write AND read-only data such as status/accounting information.

```
cmd=state file=xmodem
```



If you use the `cmd=put` or `cmd=state` to transfer V!CAS configuration files, you should also control access to these files for security reasons.

When nothing is specified, the currently selected baud rate is used for the transfer. The transfer baud rate can be changed by adding @baud to the file variable, e.g.:

```
cmd=put file=xmodem@9600 path=boot
```

Possible baud rates are 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. For transmitting data to the V!CAS (cmd=get), you should not select a rate higher than 9600. Selecting baud rates higher than the default may result in transmission errors.

In the case of transmission errors, a syslog is generated.

This feature can only be used via the SNMP shell, not via Set-up Tool.

New CAPI and TAPI Ports



BinTec product-specific TAPI and CAPI ports have been officially registered by the IANA (Internet Assigned Numbers Authority) and have been changed as follows:

	OLD PORTS	NEW PORTS
CAPI	6000	2662
TAPI	6001	2663

These default values are only used when V!CAS and BRICKware are initially configured. It was necessary to introduce these changes, because, in rare cases, there occurred conflicts with applications that used old CAPI and TAPI ports.

As a requirement for the operation of Remote CAPI/TAPI and the CAPI Tracer (PC), the values for the CAPI/TAPI ports configured on the V!CAS and the PC must be the same.

A software update on the V!CAS and on the PC does not change the configuration and with neither does it change the port numbers currently in use. Therefore, it is not necessary to change the ports after a mere update.

Nevertheless, we recommend using the new ports. In the long term, the new configuration will be necessary to resolve conflicts that may occur with NAT and Firewall configuration.

Please notice that incorrect configuration may be a potential source of errors.



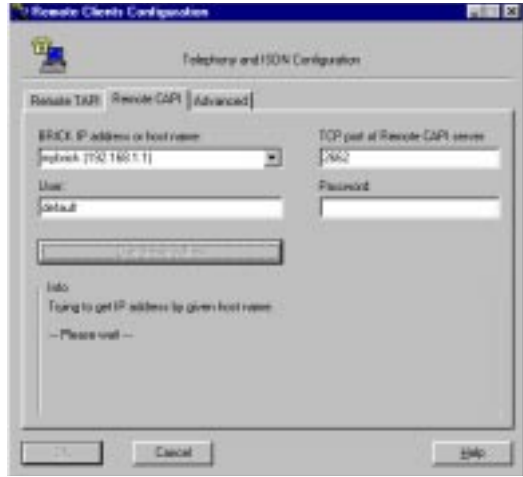
Also, ensure your changes have been saved to the boot configuration file using Setup Tool's Configuration Management menu, or using the "cmd=save" command from the SNMP shell. When only the BRICKware is newly installed or a new VICAS is taken into operation or an old one is completely newly installed, then the CAPI/TAPI ports must be adjusted manually. When VICAS and BRICKware are completely newly installed, no adjustments are necessary.

On the VICAS, the CAPI port is configured in Setup Tool in the Static Settings menu of the IP menu and the TAPI port in the Static Settings menu of the PABX menu.

VICAS Setup Tool		BinTec Communications AG
[IP][STATIC]: IP Static Settings		vicas
Domain Name	bricks.com	
Primary Domain Name Server	192.168.1.3	
Secondary Domain Name Server		
Primary WINS		
Secondary WINS		
Time Protocol	TIME/UDP	
Time Offset (sec)	0	
Time Update Interval (sec)	86400	
Time Server	192.168.1.3	
Remote CAPI Server TCP Port	2662	
Remote Trace Server TCP Port	7000	
RIP UDP Port	520	
BOOTP Relay Server		
Unique Source IP Address		
	SAVE	CANCEL

VICAS Setup Tool		BinTec Communications AG	
[PABX][STATIC]: PABX Static Settings		vicas	
Dial Procedure		Prefix # for internal calls	
Remote TAPI Server Port		2663	
SAVE		CANCEL	

On the PC, the CAPI/TAPI server ports are configured in the program “Remote Clients Configuration”. The CAPI Tracer of the DIME Tools can be configured when starting a Trace session (Start/New CAPI Trace).



The current Unix Tools “capitrace”, “eft”, and “eftd” still use CAPI port 6000 as the default setting. The ports of these programs can be changed by setting the environment variable

“CAPI_PORT” under Unix. (e.g : CAPI_PORT=2662↵, export CAPI_PORT↵)

IP Route Announcement

In the *ipExtIfTable*, there is the new variable *ipExtIfRouteAnnounce*, which adjusts for each interface under which conditions, independence of the *ifOperStatus (ifTable)* of the respective interface, the routes defined on this interface are propagated.

This new variable is relevant for the routing protocol RIP.
The variable can receive three possible values:

- ***up_only***
The routes are only propagated when the operational status of the interface is up.
- ***up_dormant***
The routes are only propagated when the operational status of the interface is up or dormant.
- ***always***
Independent of the operational status of the interface the routes are always propagated. If, for example, a dial-up interface is in the state “blocked”, the route is propagated.

For the configuration in Setup Tool see [Route Announcement](#) on page 45.

Proxy ARP

The Proxy ARP (Address Resolution Protocol) is a technique to answer ARP requests for the hardware address of a particular IP address. Normally, ARP requests are answered by the station the IP address belongs to. With Proxy ARP, the request can be alternatively answered by the V!CAS. This is useful when a host belonging to your local network is connected via WAN (e.g. a home office).

For a detailed description of the feature Proxy ARP, see Bin-Tec’s Software Reference, which is available via the WWW

Server at <http://www.bintec.de> (Section: FTP Server) from your product's page.

With this software release, the Proxy ARP feature has been enhanced. Proxy ARP must now also be configured on the destination WAN interface, through which the requested IP address would be routed.

For the LAN interface the variable ***ipExtIfProxyArp*** (***ipExtIfTable***) can receive the values off and on:

- ***off***
Proxy ARP is turned off, which is the default value.
- ***on***
Proxy ARP is turned on.

In Setup Tool, Proxy ARP for the LAN can be configured in the Advanced Settings for the LAN interface.

For the WAN interface, the variable ***ipExtIfProxyArp*** (***ipExtIfTable***) has been extended. When proxy ARP is turned on, ARP requests are answered depending on the ***ifOperStatus*** (***ifTable***) of the interface, via which the requested host can be reached. Possible values are ***off***, ***on*** and ***up_only***.

Values for ***ipExtIfProxyArp*** on the WAN interface:

- ***off***
Proxy ARP is turned off, which is the default value.
- ***on***
The request is only answered when the WAN interface has the ***ifOperStatus up*** or ***dormant***. When the interface was in the state ***dormant***, a connection is setup after the ARP request.
- ***up_only***
The request is only answered, when the WAN interface has the ***ifOperStatus up***. This value makes sense when ARP requests should only be answered in case there is already an existing connection to the requested host.

In Setup Tool, Proxy ARP for the WAN interface can be configured in the WAN Partner menu for the respective host in the Advanced Settings of the IP submenu.

The requirements for an answer to an ARP request from the LAN by the V!CAS are that the destination address be routed to a different interface from the LAN interface and that on both interfaces (LAN and destination WAN interface) proxy ARP is turned on (**on** for the LAN interface and **on** or **up_only** for the respective WAN interface). Beyond that, the **ifOperStatus** of the WAN interface must have the required state.

When you want to use Proxy ARP on a RADIUS interface, the variable **ipExtIfProxyArp** must be set via the BinTec-specific RADIUS attributes. On using BinTec-specific RADIUS attributes, see the Extended Feature Reference available via the BinTec FTP server at <http://www.bintec.de>.



Because of the extension of the Proxy ARP configuration to the WAN interface, which means additional security, the old configurations made with prior software releases are no longer compatible. To achieve the same functionality as with an activated Proxy ARP on the LAN before, the variable **ipExtIfProxyArp** must be set to **on** for the respective WAN interface.

Access Lists

The range of values the variable **ipFilterProtocol** (**ipFilterTable**) can receive has been extended. The following protocols can additionally be defined for filtering: RSVP , GRE, ESP, AH, IGRP, L2TP. (For protocol descriptions see <http://www.iana.org/>.)

In Setup Tool, the filters can be defined in the IP Access Lists menu.

X.25 Dialout Without Configuration

In an X.25 network, there are often a lot of different connection partners that cannot all be configured on the V!CAS or even on different V!CAS. In addition, there are often so many X. 25 part-

ners that a configuration is not possible because of the limited size of the flash ROM of the VICAS.

For outgoing X.25 calls, a feature was implemented which generates an ISDN number out of the destination X.25 address or the destination NSAP.

For this feature two new values for X.25 encapsulations have been added. The variable *Encapsulation* in the *biboPPPTable* and the corresponding item **Encapsulation** in Setup Tool's WAN PARTNER/ADD menu now also can receive the value *x25_noconfig* (Setup Tool: **X.25 No configuration**) and *x25_noconfig_nosig* (Setup Tool: **X.25 No configuration, No Signalling**).

The value *x25_noconfig* uses X.25 specific signalling in the D-channel for the data call.

The value *x25_noconfig_nosig* is a variation of the value *x25_noconfig* and uses, in contrast to "X.25 No Configuration", ISDN-specific signalling in the D-channel for the data call.

A detailed description of "How do I configure X.25 dial-out without configuration?" can be found in the current version of the Extended Features Reference, which can be retrieved from BinTec's FTP server at <http://www.bintec.de>.

Pools for Dynamic IP Address Assignment

Beginning in software Release 4.9 Rev. 4, it is now possible to define separate IP address pools for dynamic IP address assignments. For Internet Service Providers (ISP) and other sites with many dial-in accounts, using IP address pools is convenient for defining separate user groups. One might assign "official" addresses from one pool 1 for special accounts, and assign "non-official" addresses from pool 2 for private accounts.

At connect time, the VICAS assigns an IP address from the pool (Pool ID) defined for the respective WAN Partner. This pool ID can be retrieved from:

1. the respective partner entry in the VICAS *biboPPPTable* (using the new *biboPPPIpPoolId* variable),

2. a user record in the remote RADIUS server's users file with a BinTec-biboPPPTable="biboPPPIpPoolId=x" tag).

See the section [IP Address Pools](#) under Detailed Descriptions for additional information (including the updated Setup Tool menus).

WINS (NBNS) Negotiation over PPP

The V!CAS now supports WINS (NBNS = NetBios Name Server) Negotiation over PPP.

A detailed description of this new feature can be found in BinTec's Software Reference in Chapter 7 under the heading "DNS and WINS (NBNS) Negotiation over PPP". The Software Reference can be retrieved from BinTec's file server at <http://www.bintec.de> (Section: FTP Server). There you can find a link under "Reference Manuals" on the respective product page.

DHCP Server Functionality

The DHCP server functionality of the V!CAS has been enhanced by the features DNS (Domain Name Server) and WINS (NBNS = NetBios Name Server) Relay.

A detailed description of this new feature can be found in BinTec's Software Reference in Chapter 7 under the heading "DNS and WINS Relay". The Software Reference can be retrieved from BinTec's file server at <http://www.bintec.de> (Section: FTP Server). There you can find a link under "Reference Manuals" on the respective product page.

X.25 in Setup Tool

Two additional X.25 variables of the MIB can now also be configured via Setup Tool:



When you create a new configuration or edit a configuration in this menu, the item **L2 Window Size** can now be configured for the respective Link. The default value is 2.

This item corresponds to the variable *L2WinSize* in the *x25LinkPresetTable*.



For each routing entry, the item **Metric** can now be configured.

This item specifies a metric similar to the metric of an IP routing entry. If a call matches multiple entries in the X.25 Route Table, the routing entry with the lowest value of Metric will be used to route the call. The default value is 0.

The item corresponds to the variable *Metric* in the *x25RouteTable*.

PABX: Call Pickup

Call Pickup was implemented as a new PABX feature into the PABX of the VICAS. The Call Pickup feature needs no configuration. (All extensions configured for both POTS ports are by default assigned to one group and this setting cannot be changed.)

It is possible to make use of Directed Call Pickup and Group Call Pickup for the phones that are connected to your router.

Group Call Pickup means that when there is an incoming call on a phone which is connected to one port, you can get connected to that call from a phone which is connected to second port by lifting the handset and dialing the Group Call Pickup code.

Directed Call Pickup means that the Call Pickup code which you are dialing is directed to an incoming call on a certain extension. This can be useful, for example, when a phone can be reached via two different extensions, one for private and one for business calls. When you only want to pickup the business calls, you can direct your Call Pickup to the business call extension. (Please also see [Several Extensions for one POTS Port](#) on page 43)

If a Call Pickup is not successful, you hear the occupied signal.

The following table shows the Call Pickup codes:

Code	Function
*90#	GROUP CALL PICKUP Group Call Pickup allows you to answer a call directed to any other extension.
90<ext.>#	DIRECTED CALL PICKUP Directed Call Pickup allows you to answer a call directed to a certain extension. Here dialing " *90*# ", i.e. leaving out the extension, has the same function as the Group Call Pickup code.

Call Pickup is also possible when you are just speaking on the one line and there is an incoming call on the telephone connected to the second port:

You place the active call on hold by pressing the **R** key (or ***0#**), then dial the code for Directed or Group Call Pickup (***90#** or ***90*<ext.>#**) and you are connected to the incoming call. To return to the held call or to toggle between the calls, press the **R** key (or ***0#**) again.

Priority Voice

With this release, the Priority Voice feature is available for the products of our Voice-Data Product Line.

Priority Voice makes you reachable via the telephone connected to your router, although both B-channels are being used for data transmission to/from a WAN partner.

If both B-channels are occupied by a data connection and you want to setup an outgoing call or accept an incoming call, one B-channel of the data connection is closed down to make it available for the telephone connection.

However, you must notice that this is only possible if the two B-channels are part of the same multilink PPP connection to one WAN partner, i.e. you have configured dynamic or static channel bundling. If the two B-channels are connected to different WAN partners, the Priority Voice feature does not take effect.

Priority Voice can be configured via the SNMP shell by setting the variable ***pabxPriorityVoice*** in the table ***pabx*** to the value ***enable*** (default value: ***disable***).

The same configuration can be made via Setup Tool in the Static Settings of the PABX Menu as shown below.

VICAS Setup Tool		BinTec Communications AG	
[PABX][STATIC]: PABX Static Settings		vicas	
Dial Procedure		Prefix # for internal calls	
Remote TAPI Server Port		2663	
Priority Voice Feature		on	
SAVE		CANCEL	

The item **Priority Voice Feature** must be set to **on** to enable Priority Voice. The default value is **off**.

Credits Based Accounting System

With dial-up WAN connections, it may occur that charges increase, because of configuration errors. The Credits Based Accounting System gives V!CAS administrators the ability to control charges. It allows the V!CAS administrator to limit the number of connections, the connection time and the accounted charges of every subsystem during a specified period of time. The Credits Based Accounting System can also be used to control the PABX subsystem, i.e. the POTS ports.

You can find a detailed description of this feature in “Detailed Features Description” under [Credits Based Accounting System](#).

Bridging

Two new BinTec-specific variables are introduced to the MIB tables of the **bridge** group.

The first is the variable ***dot1dStpBridgePPPPForwardDelay*** in the table ***dot1dStp***, which allows you to time the process of the establishment of a bridging connection across the WAN.

The second variable is ***dot1StpPortBackupForIfIndex*** in the ***dot1dStpPortTable***, which is meant to determine a dialup connection as a backup connection for an existing leased line.

Both variables are described in detail below:

dot1dStpBridgePPPPForwardDelay

The unit of the value of this variable is 1/100 seconds, the range lies between 0 and 3000 and the default value is 500 (= 5 seconds).

With ***dot1dStpBridgePPPPForwardDelay*** the user can adjust how long the port of a PPP connection is waiting until a change of the port state is allowed. This concerns the changes of state from “blocking” or “disabled” to “listening”, from “listening” to “learning” and from “learning” to “forwarding”. ***dot1dStpBridgePPPPForwardDelay*** only affects WAN connections like, for example, PPP and X.25. For LAN connections further on the value of the variable, ***dot1dStpForwardDelay*** is used to determine the changes of the port state.

When the default value is used for ***dot1dStpBridgePPPPForwardDelay*** (500), it takes 5 seconds to change the state from “listening” to “learning” and another 5 seconds to change from “learning” to “forwarding”. After an ISDN connection has been established, it consequently takes 10 seconds until data is transferred in the port state “forwarding”.

This delay is necessary for the spanning tree algorithm to detect redundant paths. If there is no redundant path to your WAN connection, the value of the variable ***dot1dStpBridgePPPPForwardDelay*** can be adjusted to “0”, i.e. the port state changes immediately from “disabled” or “blocking” to “forwarding”.

dot1StpPortBackupForIfIndex

This variable is intended to be used for a setting where two BRICKs are “bridging” two LANs via PPP connections (WAN).

One connection is a leased line connection, the second a dialup connection.

To configure the dialup connection as a backup connection for the leased line connection, you must proceed as follows:

In the ***dot1dStpPortTable***, set the value of the variable ***dot1StpPortBackupForIfIndex*** for the interface of the dialup connection to the ***dot1dStpPortIfIndex*** of the leased line connection. This configuration makes the dialup connection the backup connection of the leased line connection.

As long as the leased line connection is up (***dot1dStpPortState*** is ***forwarding***), the dialup connection is not established. When the leased line connection fails, the dialup connection is established. In the latter case, when the dialup connection substitutes the leased line connection, entries in the ***dot1dTpFdbTable*** belonging to the leased line connection are deleted.

New Timer in x25LinkPresetTable

To control the state of an X.25 connection and an X.25 partner in certain time intervals, a new timer in the variable ***L2SupervTime*** has been added to the ***x25LinkPresetTable***. The value of this timer can be an integer between 100 and 30000, which is the value for the Timer in milliseconds.

Changes

TCP Optimization

TCP packets which are not confirmed are now repeated earlier. This speeds up throughput for remote CAPI, remote TAPI and Telnet.

Configuration: State File

When writing a state file with `cmd=state` the following variables are not output or substituted by "****" with Software Release 4.9.4:

- ♦ All values of the variables of *bintecsec* are not output.
- ♦ The value of the variable *AuthSecret* of the *biboPPPTable* is substituted by "****".
- ♦ The value of the variable *Secret* of the *radiusServerTable* is substituted by "****".
- ♦ The value of the variable *Secret* of the *tafServerTable* is substituted by "****".

CAPI: PLCI and NCCI

The internal process for building the values for PLCI (Physical Link Connection Identifier) and NCCI (Network Control Connection Identifier), which are used with connections between CAPI application and V!CAS, has been changed. Therefore, PLCI and NCCI are now not only unique for each application, but unique on each V!CAS.

Charging Information

Because some PABX signal charging information in the D-channel is transmitted in currency amounts, the registration of charging information on the V!CAS has been extended.

When charging information is sent as currency amounts, the charges can be read out of the variables *biboPPPConnCharge* and *biboPPPTotalCharge* in the *biboPPPStatTable* and the var-

iable ***biboPPPLinkCharge*** in ***biboPPPLinkTable***, where the charge is measured in 1/1000 of the respective currency. (E.g. receiving charging information “0.12 DM” would result in a stored value of 120 charging units.)



Please notice that when charging information is sent as currency amounts, the feature Dynamic Shorthold is not available.

When charging information is sent as units, the charges can be read out of the variables ***biboPPPConnUnits*** and ***biboPPP-TotalUnits*** in the ***biboPPPStatTable*** and the variable ***biboPP-PLinkUnits*** in ***biboPPPLinkTable***.

The PPP accounting strings in the syslog messages (info level) have changed, too. Now charging amounts and charging units are output, where charging amounts are measured in 1/1000 of the respective currency (see above). Depending on which information is signalled, one of both variables is always set.

Example:

```
16:13:17 INFO/PPP: provider: outgoing connection
closed, duration 21 sec, 10337 bytes received,
12235 bytes sent, 0 charging units, 120 charging
amounts
```

CAPI Syslog Messages

Syslog messages of the CAPI subsystem have been modified to be more informative now. CAPI now uses unique internal application identifications to make it easier to analyse debugging output.

Examples of new syslog messages:

incoming call

```
CAPI: DBG(34.023) APPL03:09 PLCI 0x0101 dialin from
<> to local number <>
CAPI: INF(34.040) APPL03:09 PLCI 0x0101 incoming
call accepted
```

outgoing call

```
CAPI: INF(371.150) APPL04:1204 PLCI 0x2E01 dialout to <>
CAPI: INF(371.172) APPL04:1204 PLCI 0x2E01 outgoing
```

call established

In these examples, APPL04:1204 or APPL03:09 identify a unique CAPI application where the first number is an application ID and the second number an internal ID, which makes it easier to assign the syslog messages to one CAPI application.

CAPI DATA_B3_IND message

CAPI DATA_B3_IND messages now contain a valid datablk counter.

Until now the datablk counter was unused and set to 0.

PABX: Idle Tone

Now two different idle tones are implemented to discern the type of dial procedure that is configured.

If the external prefix is set (external calls begin with a 0, internal calls do not have a special dial prefix), the idle tone is as follows: 3 short tones followed by a pause.

If the internal prefix “#” is set (internal calls begin with a #, external calls do not have a special dial prefix), the steady tone is the idle (dialing) tone.

PABX: Configuration during an Established Call

To use the built-in telephony services of your router or to configure some ISDN features, you have to dial special codes starting with “*”. Dialing may occur during an established connection. In such a case, to avoid the remote side also interpreting your DTMF tones as code, the connection is temporarily cut – the remote side does not receive any DTMF tones. The dialing procedure is as follows:

After dialing “*”, the second digit has to be dialed during the next 2 seconds. Then the configuration mode is active and the connection is put on hold. Now you have to dial the third digit during the next 5 seconds, otherwise the configuration mode is left. After that dial the remaining digits.

Bugfixes

isdnLoginOnPPPSDispatch

- When the variable ***isdnLoginOnPPPSDispatch (isdnTable)*** is set to allow, incoming ISDN calls with the service indicator “telephony” should be routed to the ISDN login daemon, even though the call via the ***pabxExtensionExtension*** has the matching service “PPP” in the ***pabxExtensionTable***.

For products with modem hardware, it happened that incoming calls with this signalization were dispatched to PPP routing, so that no login was possible.

This bug has been fixed.

LAPB Encapsulation with Compression

- Especially for leased line connections, it occurred that with LAPB encapsulation (IP_LAPB or MPR_LAPB) and compression (V. 42bis) data transfer was not possible. The reason was an inconsistency in the compression and decompression histories, which could result from a layer 1 disconnect. In spite of this failure, the value of the variable ***ifOperStatus (ifTable)*** was remaining “up”.

This bug has been fixed and those inconsistencies should not occur anymore.

biboPPPLQMTable

- For dial-in connections with inband authentication the interface index was not set in the ***biboPPPLQMTable*** when PPP Link Quality Monitoring was negotiated.

This bug has been fixed.

CAPI

- CAPI User Login for V!CAS

Applications using old drivers which do not perform a user login could dial out, although the user “default” was not configured in the *pabxUserTable* or a password was configured for the user “default”.

This bug has been fixed.

- CAPI User Login for V!CAS

This concerns only applications using new drivers and performing a user login when starting the program. After 256 correctly performed user logins the V!CAS rebooted. This meant that (without a reboot) an application could only be started 256 times.

This bug has been fixed.

- Direct dial-in with CAPI Applications

To receive the whole Called Party Number of an incoming call at a point-to-point ISDN interface, a CAPI application has to collect the information out of several CAPI messages it receives from the V!CAS.

For this purpose some applications only interpret those digits which are signalled to them with the “INFO_IND” message and ignore the digits in the “CONNECT_IND” message. These digits of the “CONNECT_IND” message were not sent additionally in an “INFO_IND” to the application.

Especially, when the Called Party Number was received completely in one message by the V!CAS, the application did not get any “INFO_IND” message and it occurred that in such a case a call was incorrectly accepted or not accepted at all.

This bug has been fixed.

Now all digits of the Called Party Number, which are contained in the “CONNECT_IND” message are additionally sent in an “INFO_IND” message.

- Data Transfer in Transparent Mode

When a CAPI application was sending data using the B-channel in transparent mode, it sometimes occurred that at the end of a transmission up to 31 bytes were lost.

This bug has been fixed.

Network Address Translation

- After receiving several broadcast packets via an interface where NAT is being performed, the V!CAS either “locked-up” or inadvertently rebooted. If the system locked up, the V!CAS was no longer accessible (via remote or console) and had to be switched on and off.

This bug has been fixed.

RAS and Remote TAPI (BRICKware)

- The problem occurred that when RAS (or other services) were installed under Windows NT, the TAPI could not be started. This happened because the remote TAPI wrote its configuration data to a user-specific part in the registry. When booting and before a user logged in, the PC tried to start TAPI and to read the configuration out of the default part of the registry. Therefore, the TAPI could not be started.

This bug has been fixed in the current Release of the BRICKware for Windows. Please notice that after the installation of the new BRICKware, you will have to reboot your PC twice until this change in the software is activated.

localUdpAllowTable

- When the ***localUdpAllowTable*** contained more entries than the ***localTcpAllowTable***, a reboot of the V!CAS sometimes occurred.

This bug has been fixed.

Call Collisions with MS Callback

- Microsoft Windows clients only accept incoming calls when before, via CBCP, a callback was negotiated. Sometimes a dial-out to these clients was conducted which was not negotiated as described and a call collision occurred which could cause the MS Callback to be unsuccessful.

This bug has been fixed.

ifconfig Command

- It was not possible to use the “ifconfig” command on a completely unconfigured VICAS to set the VICAS’ IP address on the LAN.

This bug has been fixed.

Now you can use e.g.

```
ifconfig en1 168.1.1.1 netmask 255.255.255.0 up
```

to configure the IP address.

Setup Tool: WAN Partner

- When configuring a new WAN partner and with that setting **IP Accounting** in the [WAN][ADD][ADVANCED] menu to **on**, the **IP Accounting** value was reset to **off**, although the menu was left with **Save**.

This bug has been fixed.

Setup Tool: Access Lists

- On a BRICK with access lists for more than 100 interfaces configured using the Setup Tool, there sometimes occurred a reboot.

This bug has been fixed.

Reboot when Establishing ISDN Connections

- In rare cases a reboot of the BRICK occurred when outgoing ISDN connections were established. The typical output with such a kind of reboot was:

```
PANIC: MIB getnext
```

```
...
```

```
or
```

```
PANIC: kmem_free: unaligned pointer
```

```
...
```

This bug has been fixed.

X.25 Routing Priorities

- The following problem occurred with X.25 connections from a BRICK across an ethernet link. When the link of the routing entry with lower metric (higher priority) was broken, the BRICK did not recognize it and nevertheless sent a CALL REQUEST to this address instead of selecting the route with the next higher metric.

This bug has been fixed by introducing a new timer in the variable **L2SupervTime** in the **x25LinkPresetTable** described under [New Timer in x25LinkPresetTable](#).

NAT on a Dial-Up Interface

- When using NAT on a dial-up interface it could occur that no more sessions were allowed, also only few active NAT sessions were opened.

This bug could be recognized when the counter **ipInAddrErrors** was counted up and no more packets were routed, although the interface was up. The problem only occurred temporarily until one connection was disconnected.

This bug has been fixed.

Setup Tool: WAN Partner Configuration

- There was a bug in Setup Tool when a number for **both** or **incoming** was configured for a WAN Partner. In such a case, the switch for **CLID** was falsely set to **no** by the system, although numbers for **both** or **incoming** were adjusted. There also occurred problems when an **outgoing** number was configured in connection with certain encapsulations. The **CLID** switch then had to be set to **yes** to make it possible to save the configuration.

All bugs which occurred in connection with **CLID** and the configuration of **incoming** and **outgoing** numbers, as well as numbers with the direction **both**, have been fixed and Setup Tool is now working correctly.

Fax: T30 Carrier detection (V.21)

- With some fax devices a change of the modulation from data carrier to V.21 command carrier was not correctly recognized. The problem increased the number of faxes that were inadvertently disconnected.

This problem was fixed using another modem operating mode that causes the modem firmware to detect the modulation change.

Accepting Calls with CAPI 1.1 Applications

- When an incoming call was accepted by a CAPI 1.1 application, the Called Party Number was returned automatically as Connected Number to the caller. Some simple PABX could not handle this information and disconnected the call.

Now the Connected Number is not sent by CAPI 1.1 applications any longer.

IPX: ripCircTable and sapCircTable

- After the command `cmd=load` had been executed, the **ripCircTable** and **sapCircTable** contained each entry twice.

This bug has been fixed.

IPX: Configuring the NetNumber

- When configuring a new WAN Partner using IPX, the NetNumber was reset to 0:0:0:0 and had to be corrected later manually.

This bug has been fixed.

HTTP Server: Internet Explorer 4.0

- When trying to access the BRICK's HTTP server with Internet Explorer 4.0 to start applications like SNMP table browsing or "htmlshow", i.e. an application which required passwords to be started, user authentication failed. The reason was that the BRICK could not recognize the HTTP protocol version 1.1, which is used by Internet Explorer 4.0, and, therefore, assumed user authentication was not possible. These applications, however, require an authentication with a password and cannot be started without one.

This bug has been fixed by enhancing the HTTP server to be able to handle all current protocol versions.

A simple workaround to solve the problem if you do not update to this software release 4.9.4:

In the Internet Explorer 4.0's **View** menu, select **Internet Options...** and in the dialog box then access the **Advanced** Tab. Here you must clear the check box "Use HTTP 1.1".

ISDN S₀: Auto Configuration

- After a reboot it seldom occurred that a VICAS at a point-to-point connection needed too long, to detect the ISDN interface via auto configuration.

This bug has been fixed.

Bridging

- **Compensation of Multiple Paths**

There was a problem with the spanning tree algorithm. In consequence of this, some ports were falsely set to **forwarding**, data packets were multiplied, and a high network traffic occurred. In rare cases the Brick rebooted.

This bug has been fixed.

- **dot1dStpPortPriority/dot1dStpPortPathCost**

To change the forwarding rules of the Brick you had to configure the MIB variables **dot1dStpPortPriority** and **dot1dStpPortPathCost**. These variables could not be adjusted however.

This bug has been fixed.

Fax Applications with Protocol Switching

- This affected few fax applications with Protocol Switching with SELECT_B_PROTOCOL_REQ. Two problems arose:
 - On sending a fax: VICAS did not confirm data transmission and the transmission was aborted by the application.
 - On receiving a fax: the sender confirmed correct transmission, although the application did not receive the fax data.

These bugs have been fixed.

CAPI and Incorrect Bearer Capability

- Incoming CAPI calls, e.g. GSM calls, with incorrect bearer capability were not signalled to the CAPI. When the bearer capability contained additional bytes to that contained in the CAPI specification, no CIP value was recognised.

This bug has been fixed.

TAPI: Calls Sent to TAPI Clients Several Times

- The PABX sent incoming calls to TAPI clients several times with different call IDs. Most Windows TAPI applications then either reported none or many calls instead of one call. In the latter case, when the TAPI application reported many calls, the user had to filter out the »right« call.

This bug has been fixed.

PPP Callback Working on the 2nd Attempt

- PPP callback was known not to work under a combination of the following conditions:
 - on the 1st attempt after a reboot
 - with user-defined numbers
 - no entry on the *biboDialTable*

This bug has been fixed.

Problems Accessing Compuserve for the First Time

- If the BRICK was newly configured, used as DHCP server on the LAN and you had configured Compuserve as your ISP, it was not possible to use a browser to establish a connection with your provider for the very first time.

In such a case, the router could not find a DNS server, necessary for connections using a name-based browser. Only connections to partners with PPP encapsulation were

made and not to partners with x.75_PPP or x.75_BT_X_PPP,
as in the case of, for example, Compuserve access.

This bug has been fixed.

Known Issues

Connection Attempts between Client and Server

- When a client tries to connect with a server but the connection is denied, for example, as a result of the client not supporting MPPE, the state of the client interface does not turn to blocked, as it does not know the reason for the failure to connect. Continued attempts to establish the connection from client to server are made.

Should the connection attempt take place in reverse (client requests MPPE, server can not fulfil the request), the state of the interface on the client side turns to blocked.

Loading of an old V!CAS Configuration via TFTP

- If a V!CAS that ran a V!CAS 4.8.6. image is updated to 4.9.4, it will not be able to read its old V!CAS configuration via TFTP transmission. The reason is that the OSPF feature is no longer contained in the V!CAS software release 4.9.4.

TFTP transmission stops when the OSPF system tables are reached. Therefore, the V!CAS configuration is not complete and this results in a disfunctionality of the V!CAS.

This known issue only concerns loading the old V!CAS configuration via TFTP transmission. The default update procedure is not influenced by this issue and can be conducted as usual.

Outgoing FTP Connections via NAT

- When outgoing FTP connections occur via NAT, data transfer does not work with some FTP servers. The connection is built up, the FTP client can register with the server. Commands such as `cd` and `pwd` work, but others such as `dir` and `get` do not.

The problem can be dealt with if the client is switched to the passive mode. This is not, however, possible with all FTP clients.

Autologout Interrupting the Update

- If the autologout time interval is less than the time it takes to install the update to the flash, the autologout occurs, interrupting the installation of the image. The flash only has time to partially write the image to the flash ROM and the update is incomplete. The procedure must be repeated.

This can occur when a rather low time interval has been set. To remedy this, either set autologout to a higher value, i.e. a time interval longer than the time it takes to install the image, or with “ $\tau = 0$ ” disable autologout completely.

Several Extensions for one POTS Port

- In general, one extension is configured for each POTS port.

For incoming calls, it is in principle possible to configure two or more extensions which are all directed to one POTS port. For outgoing calls, however, in such a case, anyone of these extensions is assigned as calling party number.

Detailed Feature Descriptions

New Setup Tool PPP Configuration Options

Global PPP Settings Menu

The new **PPP** → menu has been added to Setup Tool's main menu to allow you to configure default (non-partner specific) PPP settings. The PPP settings configured in this menu are only used when negotiating an incoming call that could not be initially identified via Calling Line ID.

VICAS Setup Tool [PPP]: PPP Profile Configuration	BinTec Communications AG vicas
Authentication Protocol RADIUS Server Authentication PPP Link Quality Monitoring	CHAP + PAP + MS-CHAP inband none
SAVE	CANCEL
Use <Space> to select	

The possible “default” PPP settings available in this menu include:

Authentication Protocol = Defines the type of PPP authentication protocol to offer the caller first. Possible values include: none, PAP, CHAP, CHAP + PAP, MS-CHAP, and CHAP + PAP + MS-CHAP.

RADIUS Server Authentication = This entry is used to configure possible RADIUS authentication on incoming calls. When set to “inband”, (the default) only inband RADIUS requests (PAP, CHAP) are sent to the defined RADIUS server. When set to “Calling Line ID”, outband requests are sent to

the server. When set to “both”, both requests are sent. Setting to “none” disables RADIUS requests.

PPP Link Quality Monitoring = Defines whether link quality monitoring is performed for PPP links. When set to “yes”, link statistics are written to the ***biboPPPLQMTable***. For detailed information about Link Quality Monitoring see page 77.

New Partner-Specific PPP Settings

Two new options, PPP Keep Alives and Link Quality Monitoring, have been added to the **WAN Partners** → **PPP** → sub-menu.

PPP Keep Alives = When this option is set the V!CAS sends LCP echo requests to the remote partner every three seconds. After five unanswered requests the PPP interface’s ***ifOperStatus*** is set to “down”. PPP keepalives is most useful (and by default, set to “on”) for leased line interfaces. The transmission of echo requests does not affect the Short Hold timer.

Link Quality Management = This option allows you to tell the V!CAS to gather PPP Link Quality statistics for a specific PPP partner. When enabled, link statistics are written to the V!CAS’ ***biboPPPLQMTable*** each time a connection is established with this partner. For detailed information about Link Quality Monitoring see Link Quality Monitoring on page 77.

Route Announcement

The Configuration option *Route Announce* has been added to the **WAN Partners** → **IP** → **ADVANCED SETTINGS** → menu. This option allows you to define under what conditions known IP routes are advertised (via the RIP routing protocols) by the V!CAS. The following three settings can be defined via this option.

- **Route Announce = up only**
When set to “up only”, an IP route is only advertised if the

operational status (value of ***ipOperStatus***) of the interface (to which the route points to) is up.

- Route Announce = up or dormant
When set to “up or dormant”, an IP route is only advertised if the operational status of the interface (to which the route points to) is either up or dormant.
- Route Announce = always
When set to “always”, the IP route is always advertised, regardless of the interface’s ***ifOperStatus***.

IP Address Pools

Pool ID Selection

When dynamically assigning an IP address to a dial-in client the address which will be assigned, or the Pool from which the address is retrieved is determined in the following order.

1. Assigning a Static IP Address

When there exists an entry in the ***ipRouteTable*** for the dial-in client, where ***ipRouteMask*** is set to a host route (= ***255.255.255.255***) and ***ipRouteType*** has the value ***direct***, in this case the IP address stored in the variable ***ipRouteDest*** of this routing entry is taken to be assigned for this WAN partner.

If caller can’t be authenticated locally via the MIB, RADIUS server(s) are contacted. If a server authenticates the caller and there is a User-Record entry

```
BinTec-ipRouteTable="ipRouteMask=255.255.255.255
                    ipRouteType=direct
                    ipRouteDest= x",
```

the IP address stored in the variable ***ipRouteDest*** of this entry is taken to be assigned for this WAN partner.

2. Assigning an IP Address from an Address Pool

If the procedures described under 1. were not successful, the IP address is assigned from the Pools.

Once the caller is identified (either inband or outband), the WAN partner’s ***biboPPPTable*** entry is compared. If the

IPAddress field = “dynamic_server” AND an address is available from the pool identified by the **PoolId** field, then a free address is assigned.

If the caller can't be authenticated locally via the MIB, RADIUS server is contacted. If a server authenticates the caller and there is a User-Record entry BinTec-biboPPPTable=“biboPPPIpAddress=dynamic_server”, the pool ID is determined from the User-Record entry BinTec-biboPPPTable=“biboPPPIpPoolId=x”.

MIB Tables Overview

Overview of new/updated system tables used in conjunction with address pools for dynamic IP address assignment.

Updated! **biboPPPTable**

Main system table for partner-specific PPP settings. Updated to include **IpPoolId** variable.

Updated! **biboPPPIpAssignTable**

Contains ranges of IP addresses that make up one or more logical address pools. Updated to include **PoolId** and **Range** variables.

New! **biboPPPIpInUseTable**

Contains entries for each address that is currently assigned/reserved. The VICAS updates the entries dynamically via the **State** field.

For detailed description of individual system table fields, please refer to the BIANCA/BRICK MIB Reference on the accompanying Companion CD or at [BinTec's WWW](#) site.

Example Configuration of IP Address Pools via Setup Tool

A. Dial-In Partner without RADIUS

IP → **DYNAMIC IP ADDRESS** → **ADD** **Create Address Pool**

First, create/modify a Pool ID to contain IP addresses that will be available for assignment at connect time.

Pool ID	1
Number	10.5.5.5
Number of Consecutive Addresses	5

WAN PARTNER → **ADD** **Create Partner Interface**

Here you'll need to set:

Partner Name	test
Encapsulation	PPP
Compression	none
Encryption	none
Calling Line ID	no

Then, in the **IP** submenu configure the V!CAS as a Dynamic IP Address server for this partner.

IP Transit Network	dynamic_server
--------------------	----------------

In the **ADVANCED SETTINGS** submenu define the Pool ID

IP Address Pool	1
-----------------	---

B. Dial-In Partner with RADIUS server

IP → **DYNAMIC IP ADDRESS** → **ADD** **Create Address Pool**

Next, modify a Pool ID to contain IP addresses that will be available for assignment at connect time.

Pool ID	2
Number	192.168.80.20
Number of Consecutive Addresses	20

Then define the following entry in the user record of the RADIUS server:

BinTec-biboPPPTable="biboPPPIpPoolId=2"

Example Configuration of IP Address Pools via SNMP Shell

In the following examples, the SNMP shell input shown in the examples A.1, A.2, and B.1 must respectively be entered in one command line.

A. Dial-In Partner without RADIUS

1. Create an IP address pool in the **biboPPPIpAssignTable**.

```
vicas:> biboPPPIpAssignAddress=10.5.5.5
biboppipAssignPoolId=1
biboppipAssignRange=5
```

2. Set the WAN partner in **biboPPPTable** to use Pool ID. Assuming entry 4 is the existing WAN partner we want to configure for Dynamic IP address assignment

```
vicas:> biboPPPIpPoolId:4=1
biboppipAddress:4=dynamic_server
```

B. Dial-In Partner with RADIUS server

1. Create an IP Address pool in the **biboPPPIpAssignTable**.

```
vicas:> biboPPPIpAssignAddress=192.168.80.20
biboppipAssignPoolId=2
biboppipAssignRange=20
```

2. Define the following entry in the user record of the RADIUS server:

```
BinTec-biboPPPTable="biboPPPIpPoolId=2"
```

3. Once the caller has been authenticated via a RADIUS server, a temporary **biboPPPTable** entry is generated. The **PoolId** field for this entry is determined by the contents of the user record discussed above.

Important Note:

Overlapping Address Pools:

Although it is legally possible to define IP address pools that overlap (as shown below), the V!CAS will not assign an address twice.

The ***biboIpInUseTable*** is consulted for this purpose.

inx	Address(*rw)	State(-rw)	PoolId(rw)	Range(rw)
0	10.5.5.1	unused	0	2
1	10.5.5.2	unused	1	2
2	10.5.5.3	unused	2	2

With the ***biboPPPIpAssignTable*** shown above, only four IP addresses could actually be used at any given time.

Credits Based Accounting System

With dial-up WAN connections, it may occur that charges rise because of configuration errors. The Credits Based Accounting System gives V!CAS administrators the ability to control charges. It allows the V!CAS administrator to watch and limit the number of connections, the connection time and the accounted charges of every subsystem during a specified period of time. If the limit is exceeded the V!CAS can not make further connections in that period of time. Syslog messages give you information about credits when the 90% or 100% mark for each limit and each subsystem is reached. Also, each time a call is rejected a syslog message is generated.

For the PABX subsystem, i.e. the POTS ports and the telephones connected to them, this feature also allows you to control incoming and outgoing connections. In this context, you should bear in mind that limiting incoming connections could result in not being able to accept incoming calls when your limit is reached.

When you adjust a maximum charge for outgoing calls of the POTS subsystem, you must consider that certain telephone companies do not transmit charging information, so that charges can not be counted.

Credits Based Accounting for the POTS subsystem will always take effect for both POTS ports. You can not configure credits for one single POTS port.

If a limit which is set for the POTS' outgoing calls is reached, you will hear the busy tone when lifting the handset. If the incoming calls' limit is reached, you also hear the engaged tone when trying to accept an incoming call, and the caller in this case first hears the ringing tone which changes into an engaged tone as soon as the called party tries to accept the call.

The new ***isdnCreditsTable*** controls this feature, it is described in the current MIB Reference at <http://www.bintec.de/download/brick/doku/mibref/index.html>.

The Credits Based Accounting System can also be configured via Setup Tool described below.

Setup Tool Menus

In the Setup Tool main menu, there are two items containing menus for the Credits Based Accounting System: **ISDN** and **Monitoring and Debugging**.

ISDN With this new item, you can manage the Credits Based Accounting System.

Monitoring and Debugging Here you can find a new menu which enables you to monitor the incoming and outgoing connections and accounted charges.



V!CAS Plus Setup Tool [ISDN][CREDITS]: Configure Credits	BinTec Communications AG vicas
Select Subsystem	
Subsystem	Surveillance
capi	off
ppp	off
isdnlogin	off
pots	off
EXIT	
Press <Ctrl-n>, <Ctrl-p> to scroll, <Return> to edit/select	

Here you can see for which subsystems accounting is active (surveillance on) or inactive (surveillance off). The default value is off. To activate accounting for a subsystem, select the corresponding item and enter the detailed settings in the next step. There are four defined subsystems:

capi

ppp

isdnlogin

pots



V!CAS Plus Setup Tool		BinTec Communications AG
[ISDN][CREDITS][EDIT]: Configure ppp Credits		vicas
Surveillance	on	
Measure Time (sec)	86400	
Maximum Number of Incoming Connections	on	
	20	
Maximum Number of Outgoing Connections	on	
	20	
Maximum Charge	off	
Maximum Time for Incoming Connections (sec)	on	
	28800	
Maximum Time for Outgoing Connections (sec)	on	
	28800	
SAVE	CANCEL	
Use <Space> to select		

Here you can enter the detailed settings for the subsystem you have selected before, here, for example, ppp.

Surveillance = Determines whether or not accounting for ppp connections is activated. If you set Surveillance on, you are able to determine the following parameters.

Measure Time (sec) = The observation interval in seconds. Enter an integer from 0 to 2147483647. Default value is 86400 seconds, which is 24 hours.

Maximum Number of Incoming Connections = The number of allowed incoming connections during the measure time. If you set it on you can enter an integer from 0 to 2147483647. Default value is off.

Maximum Number of Outgoing Connections = The number of outgoing connections allowed during the measure time. If you set it on, you can enter an integer from 0 to 2147483647. Default value is 100 calls.

Maximum Charge = The maximum charge information allowed during the measure time. If you set it on, you can enter an integer from 0 to 2147483647. Default value is off.

Charge information is measured in units or when charge information is sent as currency amounts, the charge is measured

in 1/1000 of the respective currency. (E.g. receiving charging information “0.12 DM” would result in a value of 120 charging units.)

Maximum Time for Incoming Connections (sec) = The maximum allowed time in seconds for incoming connections during the measure time. If you set it on, you can enter an integer from 0 to 2147483647. Default value is 28800 seconds, which is 8 hours.

Maximum Time for Outgoing Connections (sec) = The maximum time allowed in seconds for outgoing connections during the measure time. If you set it on, you can enter an integer from 0 to 2147483647. Default value is 28800 seconds, which is 8 hours.



VICAS Plus Setup Tool [MONITOR][CREDITS]: Monitor Credits	BinTec Communications AG vicas										
Select Subsystem <table style="width: 100%; border: none;"> <tr> <td style="width: 30%;">Subsystem</td> <td style="text-align: right;">Surveillance</td> </tr> <tr> <td>capi</td> <td style="text-align: right;">on</td> </tr> <tr> <td>ppp</td> <td style="text-align: right;">on</td> </tr> <tr> <td>isdnlogin</td> <td style="text-align: right;">on</td> </tr> <tr> <td>pots</td> <td style="text-align: right;">on</td> </tr> </table> EXIT		Subsystem	Surveillance	capi	on	ppp	on	isdnlogin	on	pots	on
Subsystem	Surveillance										
capi	on										
ppp	on										
isdnlogin	on										
pots	on										
Press <Ctrl-n>, <Ctrl-p> to scroll, <Return> to edit/select											

Here you can see for which subsystems the Credits Based Accounting System is activated (surveillance on) or not activated (surveillance off). By selecting capi, ppp, isdnlogin or pots, you can check the remaining credits for each subsystem.

Monitoring and Debugging

ISDN Credits

ppp

V!CAS Plus Setup Tool		BinTec Communications AG	
[MONITOR][CREDITS][STAT]: Monitor ppp Credits		vicas	
	Total	Maximum	% reached
Time till end of measure interval (sec)	7794	86400	91
Number of Incoming Connections	0	20	0
Number of Outgoing Connections	0	20	0
Time of Incoming Connections	0	28800	0
Time of Outgoing Connections	0	28800	0
Charge	0		
EXIT			

Here you can see the current values.

Time till end of measure interval (sec) = The seconds left in the current observation interval.

Number of Incoming Connections = The number of established incoming connections during the current measure time.

Number of Outgoing Connections = The number of established outgoing connections during the current measure time.

Time of Incoming Connections = The accounted time for incoming connections during the current measure time.

Time of Outgoing Connections = The accounted time for outgoing connections during the current measure time.

Charge = The number of charge information received during the current measure time.

Charge information is measured in units or when charge information is sent as currency amounts, the charge is measured in 1/1000 of the respective currency. (E.g. receiving charging information "0.12 DM" would result in a value of 120 charging units.)

What Was New in Release 4.8.6

Release 4.8 Revision 6

Released: 21.09.98

Features:

Bugfixes:

Detailed Description:

Contents:

Features	57
New Brickware for VICAS	57
The PABX User Concept	57
Setup Tool	58
MIB	65
ISDN Supplementary Services	65
Microsoft Callback Extension to Mode 3	67
Microsoft Callback via RADIUS	68
IPX RADIUS Extensions	69
X.25	69
IP Filter for TCP State and ICMP Type	73
New Trace Command Feature	75
Status Display for Modems	75
Wildcards for Dialing Numbers	76
Link Quality Monitoring	77
Extended Syslog Messages	77
Changes	78
D-Channel Protocol	78
Priority Voice Technology	78
Changes in Setup Tool Configuration	78
Changes in Configuration Via SNMP	82
Bugfixes	85
Known Bug	85
TP0 Bridge	85
X.31 in D-Channel	85
CAPL	86
Dynamic Shorthold	86
STAC Compression on Multilink PPP Interfaces	86
Spaces in biboPPPLoInString	87
Setting Administration Status to Down	87
Setup Tool	87
Modem	88
Detailed Feature Descriptions	89
IPX RADIUS Extensions	89
Link Quality Monitoring	92

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 [PABX]: PABX Configuration	BinTec Communications AG 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.



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

V!CAS Setup Tool [PABX] [STATIC]: PABX Static Settings	BinTec Communications AG vicas				
<table style="width: 100%; border: none;"> <tr> <td style="width: 50%; vertical-align: top;"> Dial Procedure Remote TAPI Server Port </td> <td style="width: 50%; vertical-align: top;"> Prefix # for internal calls 6001 </td> </tr> <tr> <td style="text-align: center; padding-top: 20px;"> SAVE </td> <td style="text-align: center; padding-top: 20px;"> CANCEL </td> </tr> </table>		Dial Procedure Remote TAPI Server Port	Prefix # for internal calls 6001	SAVE	CANCEL
Dial Procedure Remote TAPI Server Port	Prefix # for internal calls 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 internal calls	#	(none)
Prefix 0 for external 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.



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.


VICAS Setup Tool [PABX] [EXTENSION]: Configure PABX Extensions		BinTec Communications AG vicas	
Extension	User	Destination	
	default	isdnlogin	
	default	physical	
		physical	
ADD	DELETE	EXIT	
Use <Space> to select			

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

VICAS Setup Tool [PABX] [EXTENSION] [ADD]: Configure PABX Extensions		BinTec Communications AG 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  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 multiprotocol-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). <i>If in doubt, try this value.</i>
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

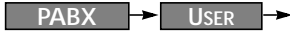
Default value: auto

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)



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**.

V!CAS Setup Tool		BinTec Communications AG
[PABX][USER][ADD] Configure PABX Users		vicas
Name	default	
Password		
TAPI Monitoring	enabled	
TAPI Controlling	enabled	
TAPI Media Streams	enabled	
CAPI	enabled	
SAVE		CANCEL
Enter string, max length = 16 chars		

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* 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* 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 78.

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 VICAS 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 V!CAS 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 *biboPPP DialTable* 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 89.

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:

<i>never</i>	No negotiation. When a call arrives that does not correspond to the default size, the call is cleared.
<i>always</i>	Negotiations are always made.
<i>when_necessary</i>	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 69.

Configuring X.25 Parameters in Setup Tool

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

• X.25 Link Configuration



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

VICAS Setup Tool		BinTec Communications AG	
[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

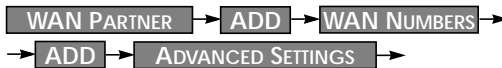


For WAN partners using the protocols X.25, X25ppp, X.31 B-Channel or X.25 no signalling the Layer 2 Mode can be configured in the advanced settings. The item Layer 2 Mode corresponds to the parameter **biboPPPLayer2Mode** in the **bi-boPPPTable**.

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

VICAS Setup Tool		BinTec Communications AG	
[WAN][EDIT][ADVANCED]: Advanced Settings		vicas	
Callback	no		
Static Shorthold	20	Idle for Dynamic Shorthold (%)0	
Delay after Connection Failure	300		
Dynamic Name Server Negotiation	yes		
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***.

VICAS plus Setup Tool		BinTec Communications AG	
[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.

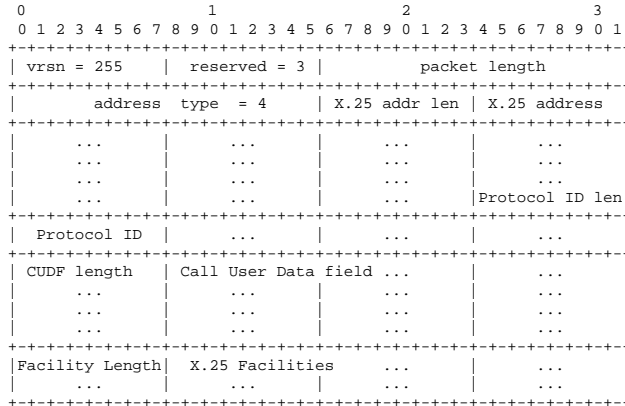
With this release two extensions concerning the transmission of X.25 data (RFC 1006) for incoming X.25 connections to the TP0 Bridge have been made.

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 bridge, 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".



VICAS Setup Tool		BinTec Communications AG
[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:



VICAS Setup Tool BinTec Communications AG [IP][ACCESS][FILTER][ADD]: Configure IP Access Filter vicas	
Description Index	TCP established 10
Protocol Connection State	tcp 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.

Hardware Interfaces

LAN	Ethernet	o.k.	
WAN	ISDN S0	unconfigured	Modem 144 used 1, available 0
LOCAL, Unit 0	Telephony	o.k.	connected RX: 12000/TX: 12000 (2048 2800)
LOCAL, Unit 1	Telephony	o.k.	used 0, available 1

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.

Wildcard for Dialing Numbers

Similar to wildcards for the calling party's address in the ***bibo-DialTable*** for incoming calls, the variable ***Number*** now can also contain wildcards for outgoing calls. The wildcards for outgoing 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 92.

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

CM-1BRI, ISDN S0 →

This menu contains settings for the ISDN interface.

VICAS Setup Tool [WAN]: WAN Interface		BinTec Communications AG 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 VICAS 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

CM-1BRI, ISDN SO →

but in the menu

PABX → EXTENSIONS →

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 1 and 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 BRICK-ware).

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 Bin-Tecs 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) Protocol(-rw) CountryCode(rw)	Descr(rw) Config(rw) AreaCode(rw)	Slot(rw) TeiProc(rw) SubscriberNo(rw)	Unit(rw) TeiValue(rw) Extension(rw)
00	0		2	0
	dss1	point_to_multipoi	automatic	0

Old Configuration:

```
vicas:isdnStkTable> isdnStkTable
```

inx	Number(*rw) Configuration(rw) TeiValue(rw)	IsdnIfIndex(rw) SPID(rw) ClearAllCalls(rw)	ProtocolProfile(-rw) TeiProc(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) Unit(rw) Layer1Prot(rw)	Type(rw) IpAddr(rw) IfIndex(rw)	Destination(-rw) User(rw)	Slot(rw) EAZ(rw)
00	"4" 0 auto	0.0.0.0 0	voice_and_data isdn_login	0
01	"3" 1 auto	voice 0.0.0.0 0	physical "user2"	3
02	"2" 0 auto	voice 0.0.0.0 0	physical "user1"	3
03	"1" 0 auto	data 0.0.0.0 0	multiprotocol_rou	0
04	"1" 0 auto	voice 0.0.0.0 0	application "faxuser"	0 "1"

Old Configuration:

`vicas:isdnDispatchTable>`

inx	StkNumber(*rw) LocalSubaddress(rw) Unit(rw)	Item(*-rw) Bearer(rw) Direction(rw)	LocalNumber(rw) Slot(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 bandwidth, the current charging intervals were not taken into consideration. This bug has been fixed.
To optimize charges now the bandwidth 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 bipoPPPLoginString

- 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 **bipoPPPTable**, 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 91.

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> = "variable1=value1 ... variablen=valuen"
```

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 ***biboPPTable*** 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 ***biboPPTable*** or via RADIUS server) the variable ***biboPPProfileLQMonitoring*** in the ***biboPPProfileTable*** must be set to on.

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

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

biboPPPLQMTTable:

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>.