RELEASE NOTE
BIANCA/BRICK-XM
July 15, 1997

New System Software:
Release 4.4 Revision 8

This document describes the new features, enhancements, bug-fixes, and changes to the BIANCA/BRICK-XM System Software since Release 4.4.

All user documentation is available in Adobe's PDF format via BinTec's HTTP server at http://www.bintec.de.

Upgrading System Software

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

2. With this image you can upgrade the BIANCA/BRICK-XM 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 *BRICK-XM User's Guide* under *Firmware Upgrades.*

3. Once you've installed release 4.4 Revision 8 you may want to retrieve the latest documentation (in Adobe's PDF format) which is also available from BinTec's FTP server 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 FTP server.

What's New in Revision 8

4.4 Revision 8:   Released: 15.07.97

Features:

Access List Extensions

Access Lists have been extended to include support for:
- Port Ranges for IP Access Lists
- Access List Violation Actions
- Access List Violation Reporting

Port Ranges for IP Access Lists

Port Ranges can now be configured for Access Lists (Allow and Deny lists) making it easier to apply an access list to a range of ports.

Configuration

In past releases access lists had to be configured for each restricted port. Using the new ***PortRange*** variables, an access list can be applied to match a complete range of ports. By default these variables are set to "-1"; when set to any other value the respective access list is extended to match all packets within the range of ports. Note that the ***PortRange*** variables define the last port number in the range (and not the total number of ports).

New Variable	Extends Port Range to:
ipAllowSrcPortRange	*ipAllowSrcPort* $\leq$ Range $\geq$ *ipAllowSrcPortRange*
ipAllowDstPortRange	*ipAllowDstPort* $\leq$ Range $\geq$ *ipAllowDstPortRange*
ipDenySrcPortRange	*ipDenyDstPort* $\leq$ Range $\geq$ *ipDenySrcPortRange*
ipDenyDstPorlfRange	*ipDenyDstPort* $\leq$ Range $\geq$ *ipDenyDstPortRange*

Port Ranges can easily be configured via Setup Tool in the `IP` → `ACCESS LISTS` menu, by selecting "`specify range`" in the **Specify Port** field.

Access List Violation Actions

In previous releases the BRICK always refused IP packets that were restricted by a configured IP access list (*ipDenyTable*) by sending an "ICMP Destination Unreachable" message to the sender. The new *ipExtIfAccessAction* variable (*ipExtIfTable*) defines the default action to take when an Access List is breached.

Configuration

ipIfExtAccessAction may be set to:

ignore   The BRICK simply discards packets (default).

refuse   The BRICK discards the packet and transmits an "ICMP Destination Unreachable" message to the sender.

Access List Violation Reporting

With Access List reporting you can now gather information about security breaches on the BRICK. Each time the BRICK receives a packet the configured access lists are applied. If a packet is restricted by a matching access list a brief report can be generated.

Violation Reports are generated via syslog and are saved in the *biboAdmSyslogTable*. This makes it possible to save Access List reports to remote loghosts. By default, reporting is disabled.

The new *ipExtIfAccessReport* variable (*ipExtIfTable*) defines how the BRICK reports access list violations and may be:

info   A syslog message containing the protocol, IP addresses, and port numbers is generated in the syslog table (default value).

dump   In addition to the same report generated for info the first 64 bytes of the IP packet are dumped (in hex format) to the syslog table.

none   No reports are generated.

NOTE: **Logging Access List Reports to remote log hosts via syslog**. Reports use level = info, facility = error.

Dynamic Name Server Address Resolution

Starting with Revision 4 the BRICK can be used as a DHCP server. DHCP clients that request the DNS server's IP address from the BRICK are always given the current address set in **biboAdmNameServer** if one is configured; otherwise the value of **biboAdmNameServ2** is used. If neither variable is set the BRICK sends it's own IP address and attempts to resolve the name server's address dynamically (see below) using <u>DNS Negotiation over PPP</u> after the first resolution request is received.

DHCP Clients　　　　　**DHCP Server**

biboPPPTable
IpAddress = `dynamic_client`

Windows 95

DHCP_Req
NAME_SERVER

❶ DHCP_Resp
192.168.5.1

192.168.5.1

192.168.12.1

❺

❷ resolve
ftp.brick.com

Dynamic Resolution

ISDN

❸ "DNS Server temporarily down ..."

ftp.brick.com
192.168.5.45

192.168.0.1

Windows NT

DHCP_Req
NAME_SERVER

❹ DHCP_Resp
192.168.0.1

time
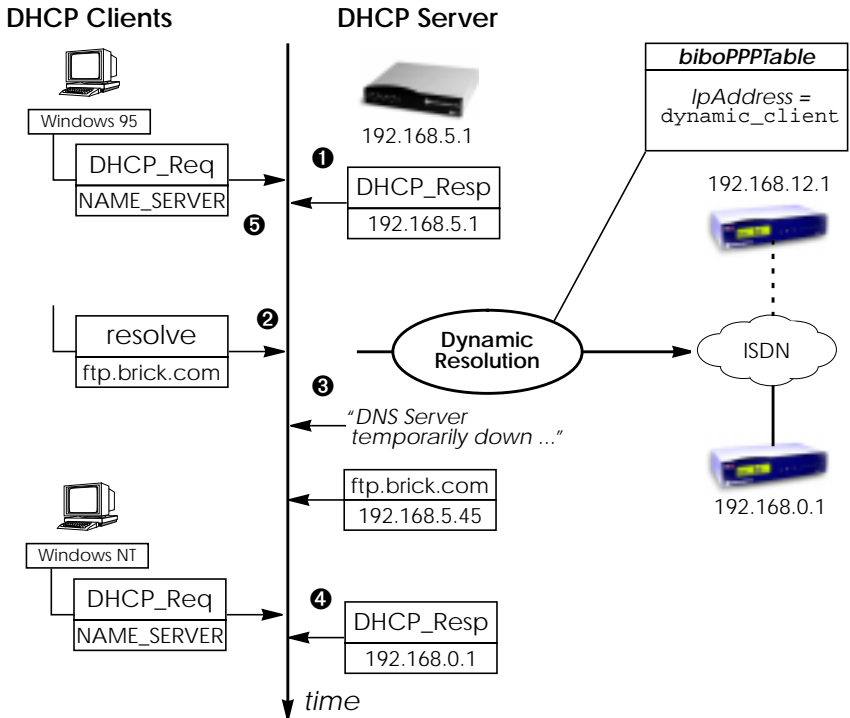
❶ If no DNS server is configured, the BRICK sends it's own IP address.

❷ Upon receipt of first name resolution request, the BRICK parses the **biboPPPTable** for partners that support DNS negotiation; i.e., **IpAddress** field is set to `dynamic_client`.

❸ While attempting to configure it's DNS server, DNS requests are answered with "DNS Server temporarily down".

❹ Once the DNS's address is successfully negotiated, the BRICK can inform subsequent DHCP requests for a name server with it's newly configured address.

❺ Clients that were given the BRICK's address as name server can't be informed of the "new" address. For these hosts, the BRICK simply continues relaying resolution requests to the actual DNS server.

Special IP Interfaces

In previous releases two special IP interfaces (*ifIndex* 0 and 1) were available on the BRICK. Beginning with revision 8, a third interface, the "IGNORE" interface, (*ifIndex* 2) has been added. These special interfaces are now listed in the BRICK's *ifTable*.

inx	IfTable Index	Meaning
00	*0 (REFUSE)*	When packets are routed to this interface, the packet is discarded and an "ICMP Destination unreachable" message is sent to the sender.
01	*1 (LOCAL)*	Packets routed to this interface are given to an appropriate internal process on the BRICK.
02	*2 (IGNORE)*	Packets routed to this interface are discarded. No response is sent to the sender.

TIP: Special IP interfaces are useful for Extended IP routes. For example an extended IP route could be used to routes all DNS requests received from a specific host to the IGNORE interface.

Van Jacobson Header Compression

The BRICK now supports Van Jacobson TCP/IP header compression (VJHC) according RFC 1144. TCP/IP header compression is a method used to reduce the size of TCP/IP packets and provides improved performance (line efficiency) for dialup connections.

Configuration

VJHC can be configured for selected WAN partners via Set-upTool or the SNMP shell.

• Setup Tool

In the WAN PARTNER ➜ ADD ➜ ADVANCED SETTINGS menu, select either on or off in the "Van Jacobson Header Compression" field.

• SNMP Shell

In the ***biboPPPTable***, set the new *VJHeaderComp* variable to either enabled or disabled.

TIP: VJHC is required for accessing Deutsche Telekom's T-Online via PPP in Germany.

New PMX Driver

A new PMX driver has been implemented allowing for improved overall throughput with primary rate interfaces. The system overhead required for dynamic switching of channels has also been greatly reduced.

Except for the new ***pmxIfPortMode*** variable, the new PMX driver is backwards compatible with older CM-PRI modules (***biboABrdHWRelease*** < 2.1).

Layer 1 Error Conditions

The ***pmxIfTable*** has been extended and consists of several new variables to help detect layer 1 error conditions for primary rate interfaces. All of the variables described below are counters that keep track of the number of times a specific error condition has occurred since the last system boot.

• *pmxIfLOS*

Loss Of Signal conditions since startup.

• *pmxnoIfnoFAS*

Loss Of Frame Alignment conditions since startup.

• *pmxIfRemoteAlarms*

Remote alarms received since startup. Remote alarms are transmitted by the switching station when too many framing errors have beendetected.

- *pmxIfAlarmInds*
  Alarms received since startup. These alarms are transmitted when less than 2 null bits are received within 250μ.
- *pmxIfCRC4Errors*
  CRC4 errors detected since startup. The CRC-4 method is used to ensure frame synchronisation.
- *pmxIfPositiveSlip*
  Positive Slip conditions detected since startup; flags that the frequency of the receive clock is less than the frequency of the internal system clock
- *pmxIfNegativeSlip*
  Negative Slip conditions detected since startup; flags that the frequency of the receive clock is greater than the frequency of the internal system clock.

NOTE: Some of these variables will occaisionally change during normal operation and when booting until the BRICK has synchronized with the switching station. Continual changes however may indicate a problem with the primary rate interface.

BIANCA/CM-PRI "OUT" Port

Sites using hardware version 2.1 of the BIANCA/CM-PRI module can now set the OUT port's mode via the new ***pmxIfPortMode*** variable. The OUT port allows a connected PMX line to be automatically switched over to a backup router.

pmxIfPortMode can be set to:

`normal`	The OUT port is disabled. (Default mode).
`go-thru`	The OUT port is enabled. In this mode signals received on via the IN port are redirected to the OUT port. This condition also results when the BRICK loses power.
`go-thru-tst`	The same as `go-thu` mode except that Layer 1 remains active.

Technical data including pin assignments, cabling requirements, and LED indicators is described in the Release Note (dated April 7, 1997) accompanying the BIANCA/CM-PRI module (also available via BinTec's FTP server).

Wildcards for Calling Party's Address

Wildcard characters can now be used in the *Number* field of the **biboDialTable** to match different Calling Party's Numbers at connection time. (The same wildcards are already supported in X.25 in the **x25RouteTable**.) Wildcards may also be used from Setup Tool in the ▎WAN PARTNERS ▎➤▎ISDN NUMBERS ▎ menu.

This means you don't have to configure separate Dial Table entries for each MSN your partner may be calling from. The table below lists the currently supported wildcards.

biboDialNumber Wildcard Matching

*	Match zero or more digits. 45* matches any number beginning with 45, i.e., 45, 4512, 4512345, 459, etc.
?	Match any single digit. 5? matches 50 through 59.
[ ]	Brackets denote a set of possible digits to match. A hyphen may be used for inclusive ranges. 21[45] only matches 214 or 215 (4 or 5) 21[6-8] matches 216, 217, 218 (6 through 8, inclusive) 21[^9] matches 210 through 218. (not 9)
{ }	Curly braces denote an optional string to match. {0911}2145 matches 09112145 and 2145 (optional)

If the Calling Party's Number from the incoming call matches a **DialTable** entry with wildcards and an entry without wildcards, the entry without wildcards is always used.

NOTE: Configuring wildcards from the SNMP shell.
A *DialNumber* containing a **?** must be quoted using double quotes (**"**) to avoid the SNMP shell from interpreting the character as a help command.

```
mybrick:biboDialTable > biboDialNumber:05="09115678?"

05: biboDialNumber.10001( rw):      "09115678?"

mybrick:biboDialTable >
```

Enhancements

PPP Connection Optimization

Most remote access routers accept incoming data connections even after Outband identification (Calling Line ID) was unsuccessful to allow for Inband authentication (CHAP and/or PAP). A common problem posed by this mechanism is that ISDN charges are needlessly incurred for the caller because most sites use Inband authentication as a supplement to CLID.

For this reason the mechanism for accepting PPP connections on the BRICK has been optimized as follows.

Incoming PPP connections are no longer accepted when:

1. No WAN partners are configured (***biboPPPTable*** is empty)[1], OR
2. All partners are configured for CLID and the incoming call doesn't match any partner's number (***DialNumber*** entry).

CLID is configured in SetupTool's   WAN PARTNER   menu by setting the "Identify by Calling Number" field to on.

Provider Setup via SetupTool

Setup Tool's   PROVIDER CONFIGURATION   menu (under WAN partners, see <u>Rel. 4.4 Rev. 6</u>) has been enhanced and can be used to configure access to online service providers shown below.

Online Provider	Encapsulation from Setup Tool
Compuserve via T-Online	async PPP over X.75/T.70NL/T-Online[b]
Compuserve Corporate Network	async PPP over X.75[a]
	async PPP over X.75/T.70NL/T-Online[b]
Compuserve Network	async PPP over X.75[a]

a. For direct access (***biboPPPEncapsulation***=x75_ppp).
b. For indirect access via the T-Online gateway
     (set ***biboPPPEncapsulation***=x75btx_ppp from the shell).

———————————————

1. and a RADIUS server (***biboAdmRadiusServer***) isn't configured.

NOTE: When configuring the online providers shown
        above, you must select the respective encapsu-
        lation from Setup Tool.
        i.e.: ***biboPPPEncapsulation*** from the SNMP shell

Terminal Adapter Support

The default MRU/MTU value used during LCP negotiation has
been changed from 2048 to 1524. Although 2048 bytes is an IETF
standard, many devices, in particular older TAs (Terminal
Adapters), aren't compatible to the this standard. This means
the BRICK supports even more Terminal Adapters than before.

Hyperchannel Support

Sites using the BRICK-XM with the CM-PRI module can now
configure hyperchannels consisting of a single B-channel (no D-
channel) with a maximum bandwidth of 1984 Kbps.

The configuration involves setting the *ChannelMode* object
(in the ***pmxIfTable***) to `e1_burst`.

Setup Tool

An internal cache has been implemented within Setup Tool that
greatly reduces the time required to load and manipulate large
configuration files (configurations with more than 100 dialup
partners).

Bugfixes

Accounting Information
- The *biboPPPTotalUnits* and *biboPPPConnUnits* fields (***bi-
  boPPPTable***) are now correctly set. In previous releases
  charging information was displayed correctly only in the
  PPPLinkTable, ***isdnCallTable*** and ***isdnCallHistoryTable***.

CM-EBRI
- With certain telephones the BRICK's CM-EBRI didn't al-
  ways acknowledge DTMF tones. This is because some tel-
  ephones (regardless of how long the buttons are pressed)

send very brief DTMF signals. To allow for greater compatibility the minimum time required to acknowledge DTMF signals has been reduced from 100 ms to 20 ms.

PPP

- Response packets (access list restrictions) and other packets that can't be routed no longer affect the **ShortHold** timer for PPP connections.

TCP

- Under certain circumstances an initial connection to a particular TCP service (telnet, capi, rfc1006, http) hindered subsequent connections to the same service. Previously, this problem could only be resolved by rebooting the system. This problem has been corrected in revision 8.

- A protocol problem hindered the BRICK from accepting TCP connections from some UNIX systems (i.e. SVR4). This has been corrected.

ISDN

- Previously, ISDN connections with 1TR6 PBXes that weren't operating in conformance to 1TR6 were closed. In particular, when an empty Cause W-Element was sent in the STAT message. Since many devices exhibit this behaviour the 1TR6 protocol on the BRICK has been adapted to be more tolerant.

- The BRICK is much more stable when used in connection with ISDN leased line configurations.

- The **NumberOfChannels** object in the **isdnIfTable** is now correctly set for ISDN hyperchannels.

- The pattern matching function (used to match the calling party's number of an incoming call) didn't always work correctly when the ***** wildcard character was used. Wildcard characters and the pattern matching function now work correctly.

Changes in Previous Revisions

4.4 Revision 6:   Released: 03.06.97

Features:

Dialup Access to CompuServe Online Services

To allow for dialup connections to CompuServe Online Services two additional encapsulation methods have been added to the *biboPPPEncapsulation* variable:

`x75_ppp`	async PPP over X.75
`x75btx_ppp`	async PPP over X.75/T.70NL/T-Online

These settings can be used to enable the BRICK to dial into a CompuServe Network Node directly (x75_ppp) or to access CompuServe indirectly through T-Online's CompuServe Gateway (x75btx_ppp).

CIS Configuration with Setup Tool

In WAN PARTNER → ADD → you'll need to set:

Partner Name	cis
Enabled Protocols	IP
Encapsulation	Async PPP over X.75
Identify by Calling Number	yes
PPP Authentication Protocol	none

Then, under ISDN NUMBERS → set:

ISDN Number	*<CSI's telephone number>*
Direction	outgoing

Under IP → assign dynamic to transit network field.

IP Transit Network	dynamic

Under ADVANCED SETTINGS → PROVIDER CONFIGURATION →

Provider	Compuserve Network
Host	CIS
User ID	*<your CIS member ID>*
Password	*<your CIS password>*

This information is used to generate the ***biboPPPLoginString*** variable. See <u>Using the biboPPPLoginString</u> (page 14) for examples.

13

TIP: When accessing CompuServe through the T-Online Gateway using "Async PPP over X.75/T.70/BTX" make sure to use the ISDN number: 01910 to ensure local charging tariff.
Also, you may want to set *ShortHold* to 100 since the CIS login may take up to 20 seconds or more.

PPP Login chat-script functionality

The **biboPPPTable** has been extended to include an additional *biboPPPLoginString* variable which can be used to control login sessions to server systems. The LoginString consists of user definable expect – send sequences comparable to chat scripts commonly used on other systems.

Using the *biboPPPLoginString*

After the initial ISDN connection is established a login sequence can be invoked with the help of the *biboPPPLoginString*. Upon successful completion of the login sequence, the BRICK converts the link to an asynchronous PPP connection.

Syntax

biboPPPLoginString is a quoted string consisting of special characters and alternating expect – send sequences separated by spaces. The first string detected as not being a special character is assumed to be an expect string.

The following special characters are currently recognized:

-d<*number*>	Indicates a pause of <*number*> seconds.
\n	Indicates transmit one carriage return.

Examples

```
"-d1 \n ogin: user sword: secret"
```

Setting *biboPPPLoginString* to the above setting results in the following login sequence:

Wait 1 second	
transmit:	\n
expect:	ogin:
transmit:	user
expect:	sword:
transmit:	secret

Note: When SetupTool is used to configure a dialup connection to CompuServe the appropriate LoginString is generated automatically according to the information (Host, User ID, and Password) entered in the `Provider Configuration` menu.

For a direct connection to CompuServe the following string could be generated (see CIS Configuration with Setup Tool p. 13).

"-d1 \n e: **CIS**\n D: **12345,6789**/go:pppconnect\n wor -d1 **secret**\n PPP"

For access through T-Online, this string might be generated.

".\n :000000 000327278259\n gabeseite 11 # # Name: **CIS**\n ID: **12345,6789**/go:pppconnect\n wor -d1 **secret**\n PPP"

ISDN Login Screening

The *isdnloginAllowTable* has been added to the isdn subsystem and gives you additional control over which callers may access the isdnlogin service on your BRICK.

Background

The calling party's number (CPN) reported by an incoming call may be assigned by the user placing the call or by the telephone switching station. If the CPN was assigned by the user the switching station may optionally verify the address. The party that assigned the CPN and whether or not the CPN has been verified is reported in ISDN in the Screening Indicator field of the call packet.

Usage

Based on the information reported in the incoming call packet, the *isdloginAllowTable* fields, in particular the *Screening* field, can be used to gauge the "trustworthiness" of the CPN field. Since incoming calls are initially routed according to the *isdnDispatchTable* configuration, these rules only apply to calls dispatched to the isdnlogin service.

The *isdnloginAllowTable* consists of 4 fields as follows:
• *isdnloginAllowStkNumber*
  The ISDN stack the incoming call arrived on. The value "-1" can be used to match all stacks.

15

- *isdnloginAllowRemoteNumber*

  The remote number (Calling Party's Number). The following wildcard characters may be used to match multiple CPNs.

*	Match zero or more digits. 45* matches any number beginning with 45, i.e., 45, 4512, 4512345, 459, etc.
?	Match exactly one digit. 5? matches 50 through 59.
[ ]	Brackets denotes a set of possible digits to match. 091121[45] only matches 0911214 or 0911215. 091121[6-8] matches 0911216, 0911217, 0911218. 091121[^9] matches 0911210 through 0911218.
{ }	Curly braces denote an optional string to match. {0911}2145 matches 09112145 and 2145.

- *isdnloginAllowRemoteSubaddress*

  The caller's remote subaddress.
- *isdnloginAllowScreening*

  The minimum trust-level. This means accept calls with an indicator greater than or equal to this level. Indicators are ordered from highest to lowest as follows:

Screening (*isdnloginAllowTable*)	CPN assigned by	Verification Status
network	network	not attempted
user-verified	user	verification successful
user	user	not attempted
user-failed	user	verification failed
dont_care	—	—

Example

The ***isdnloginAllowTable*** fields are additive; that is the characteristics of an incoming call must match all fields of an entry before the call is accepted.

This example configures the BRICK to "Only accept calls..."

1. ...received on stack 0.
2. ...that have been verified or set by the ISDN.

```
mybrick:isdnloginAllowTable > StkNumber=0
    RemoteNumber={0911}214[46]    RemoteSubaddress=-1
    Screening=user-verified
```

3. ...from (0911) 2144, and (0911) 2146 (with or without the area code, here 0911).

PPP Screening

A new *Screening* field has been added to the **biboDialTable.** This field is compared to the Screening Indicator of an incoming call (See Background). This gives you an additional measure of control for verifying incoming calls from configured partners. By default, *Screening* is set to "dont_care" but may optionally be set to network, user, user-verified, or user-failed.

The *biboDialScreening* field works just like the *isdnlogin-Screening* field and only applies to incoming calls, i.e, where *biboDialDirection* = both, or incoming.

DNS Negotiation over PPP

The BRICK now supports DNS (Domain Name Service) Server Negotiation over PPP according to RFC 1877. This allows the BRICK to request a valid IP address for its primary/secondary name server at connect time depending on the value of the *biboPPPIpAddress* field (dynamic_client or dynamic_server) for the PPP partner.

Note that primary and secondary addresses are negotiated independently. DNS negotiation is handled at connect time (IPCP/NCP layer of PPP) and generally proceeds as follows.

Scenario 1:   *biboPPPIpAddress*=dynamic_client
The local peer requests a valid address for primary/secondary DNS server and sets *biboAdmNameserver* and *biboAdmNameServ2* if values are provided by the remote peer. If these values were already assigned, they will be overwritten.

Scenario 2:   *biboPPPIpAddress*=dynamic_server
The local peer does not request a primary/secondary DNS server address. If the remote peer requests these addresses,

17

the local peer sends the current values of *biboAdmName-server* and *biboAdmNameServ2* to the remote peer.

Enhancements

ISDN Screening Indicator

The *isdnCallTable* and the *isdnCallHistoryTable* contain a new *Screening* field. This field stores the Screening Indicator reported by the ISDN for each incoming call. The new ISDN Login Screening and features utilizes this ISDN service for controlling access to the isdnlogin service on your BRICK.

ISDN Calling Information

The *isdnCallTable* and the *isdnCallHistoryTable* contain a new *Info* field. This field stores additional information about the current (or closed) ISDN call. For CAPI calls, the Info field contains a string identifying the IP address and port number of the CAPI host the call is associated with. (i.e.: "CAPI 199.1.1.5:1094"). Calls dispatched to the idsnlogin service are identified with "isdnlogind" (incoming calls) and "isdnlogin" (outgoing calls).

AccountingTemplate

This information can be included in accounting messages by assigning the "%I" identifier to the *isdnAccountingTemplate*. Note that by default the "%I" identifier is not included in the template but is available as of Revision 6.

Process Priorities

The BRICK's internal process priorities have been adjusted to optimize routing and to allow configuration sessions and remote access even during times of high system loads. The relative priorities of the internal processes are as follows. .
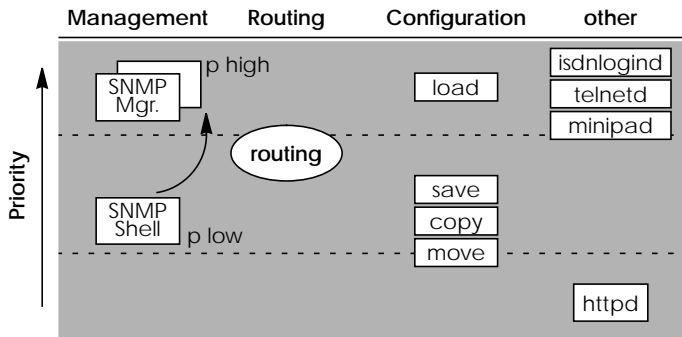
Note: As of release 4.4 Revision 4, the priority of the SNMP shell (and the subprocesses it starts) can be set to high using the `p` command. In Revision 6 the default level is low.

Shell-priority and `cmd=save` commands:

If shell-priority is set to `high` and a configuration is saved, the shell immediately returns you to the command prompt. (In contrast to `low` status where the prompt is returned only after the save command is completed.) The state of a cmd=save command can be verified by displaying the ***biboAdmConfigTable***.

Note that this does not apply to SetupTool sessions. SetupTool always waits for configuration commands to complete before proceeding.

Setup Tool

The ▐ WAN Partner ▌ menu has been extended and can now be used to configure dialup IP connections to CompuServe Online Services. The new ▐ Provider Configuration ▌ menu contains user access information (host machine, member ID, and password) which is used to generate *biboPPPLoginString* (see Using the biboPPPLoginString, page 14) used at connection time.

```
BIANCA/BRICK-XS Setup Tool                    BinTec Communications GmbH
[WAN][EDIT][ADVANCED][PROVIDER]: Provider Configuration(cis)      mybrick


    Provider                         Compuserve Network

    Host                             CIS
    User ID                          12345,6789
    Password                         secret




                    OK                        CANCEL

 Use <Space> to select
```

Provider = Defines the type of access to CompuServe and may be one of the following:
- Compuserve Network
- Compuserve via T-Online
- none

Host = The CompuServe hostname to dial into.

User ID = The user's CompuServe Member ID to use for the connection.

Password = The password to use for the User ID specified above.

Token Ring

Bridged LAN source-routing operation by end systems is now supported. MAC frames transmitted by the BRICK now include an additional Routing Information Field. This field consists of a list of source-routing bridges that lie between the BRICK and the destination. This allows intermediate bridges that support source-routing but not transparent-bridging to relay MAC frames sent by the BRICK.

New Proxies

A new proxy for VDOLive Audio and Video Streaming has been implemented. PCs on NAT protected/hidden networks can now connect to VDOLive services provided by external hosts.

Bugfixes

BootP/DHCP

- The BootP daemon sometimes used system resources even when DHCP was disabled. This has been corrected.

PPP

- The *biboPPPTotalUnits* field now takes PAP/CHAP/Radius authentication failures into consideration.

- Broadcast (RIP) packets no longer affect the ShortHold timer for PPP connections.

21

- In previous releases, temporary interfaces (created by ra-diusd) were incorrectly saved during a **`cmd=save`** command. These interfaces are no longer saved.

X.25

- A system panic occurred on some systems in connection with the the BRICK's local X.25 interface (i.e. rfc1086 or minipad) and has been corrected in revision 6.

4.4 Revision 5:   Released: 03.06.97

Features:

X.25 over Dialup ISDN

A new PPP encapsulataion method was added to the ***biboPPPTable***. The value **`x25_nosig`** (no signalling) can be used to allow outgoing calls to be placed over an ISDN dialup line.

   With **`x25_nosig`** encapsulation outgoing calls are not signalled as X.25 calls (as with existing **`x25`** encapsulation) but as a data transfer call (DSS1: Bearer Service unrestricted digital info without LLC).

Enhancements

X.25 over Ethernet

The BRICK now supports the XID procedures specified in ISO 8802-2 (Logical Link Control). In addition the BRICK also responds to TEST packets.

   The MAC address format defined in the MIB has been extended to support addresses that include a 7th octet. The 7th octet can be used for the Remote-SAP of the LLC connection.

Setup Tool

- As of revision 4, Setup Tool menus containing many list entries now offer a scrollbar that displays the begin and end of a list. The PageUP and PageDown keys (or Ctrl-B and Ctrl-F) can now be used to scroll through list entries.

Bugfixes

Ethernet/Token Ring

- The BRICK sometimes paniced while booting when an LLC SABM packet had already been received via an ethernet or token ring interface.

4.4 Revision 4:   Released: 6.05.97

Important Changes

NOTE: Beginning with Release 4.4 Revision 4 the routing algorithm used for selecting IP routes with respect to default routes has changed.

Previously, a default route (*ipRouteMask* = 0) was used to route packets when a respective network or host route was in the "blocked" or "down" state.

Beginning with Revision 4, default routes are used when an appropriate network or host route is not found. If a default route is also unavailable IP packets are discarded and a "Destination Unreachable" message is sent.

Features:

DHCP (Dynamic Host Configuration Protocol)

The BRICK can now be used as a DHCP server allowing it to assign IP addresses making it easy to manage a limited amount of IP addresses for a large number of local or remote DHCP clients.

Client machines (Win 95/NT) that support DHCP are generally configured to retrieve their IP address from the server and adjust their configuration's appropriately. With DHCP the retrieved IP address is only valid for a specified time period, known as the "Lease Time". Once the lease time has run out, the server is free to reassign the IP address when needed. The DHCP server also informs clients of the appropriate nameserver (*biboAdmNameServer* is used) and default gateway.

Configuration with Setup Tool

To configure DHCP see the [IP] ► [DHCP SERVER] menu.

Interface	IP Address	Number	Lease Time (Minutes)
en1	199.1.1.70	15	30
en1	199.1.2.25	5	120
tr3-snap	200.1.2.50	4	120

ADD   DELETE   EXIT

Interface = Associates a BRICK interface with a set of IP addresses. The BRICK will assign an available IP address from the appropriate set of addresses depending on which interface it received the address-request on.

IP Address = Defines the first IP address in the set.

Number = Defines the number of addresses in the set (including the first address).

Lease Time = Defines the time in minutes addresses from this set are valid. Addresses become available for reassignment once the lease time runs out.

Internet Access for the LAN using DHCP and NAT

DHCP can be used in combination with NAT (Network Address Translation) to provide easy Internet access for a complete LAN. The main advantage is that PCs on the LAN don't need to be configured individually.
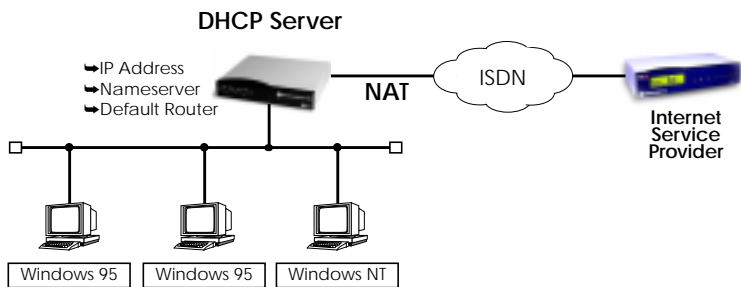
DHCP Clients

A simplified configuration using this setup would involve:

1. Configuring Network Address Translation on the BRICK (only one official IP Address is required).

25

2. Configure BRICK as DHCP Server.

New SNMP Shell Commands

Two new commands (p and ifconfig) have been added to the SNMP shell and are described below.

New Shell Priority Command

The **p** (priority) command sets the priority (high or low) of the BRICK's SNMP shell with respect to other routing processes. The syntax for the command is as follows:

$$\text{\textbf{p} [\textbf{high} | \textbf{low}]}$$

The specified priority becomes effective for the current shell and all sub-processes started from this shell. If no options are specified, the current priority is displayed.

By default, the SNMP shell has a higher priority than routing processes which means that an interactive configuration session (setup) could affect performance on systems with many WAN partners.

New *ifconfig* command

The **ifconfig** command can be used to assign an address to a network interface and/or to configure network interface parameters and changes the respective routing table entries.

When only the required interface parameter is used, ifconfig displays the current settings for the interface.
The syntax for the command is as follows:

ifconfig <*interface*> [**destination** <*destaddr*>] [<*address*>]
  [**netmask** <*mask*>] [**up** | **down** | **dialup**] [**-**] [**metric** <*n*>]

Options and their respective *ipRouteTable* entries are as follows:

<*interface*>	Interface name (ifDescr)
destination <*destaddr*>	Destination IP address of a host for adding host routes. (ipRouteDest, ipRouteMask)
<*address*>	BRICK's IP Adress for this interface. (ipRouteNextHop)
netmask <*mask*>	Netmask of interface (ipRouteMask)

–	Don't define own IP address (i.e. ipRouteNextHop = 0.0.0.0)
metric <*n*>	Sets route metric to n (ipRouteMetric1)

Enhancements:

Enhanced Time Service Support

- In previous releases the BRICK was only able to retrieve current time from a host via UDP. This mechanism has been expanded so that the current time can be retrieved from any of the following four methods. Also, the interval at which current time is retrieved can be set.
  - Time Service (RFC 868) via UDP
  - Time Service (RFC 868) via TCP
  - Simple Network Time Protocol (RFC 1769) (via individual Time Requests or Broadcasts)
  - ISDN D-channel (**stack 0 only**)

An overview of the relevant SNMP variables are as follows.

biboAdmTimeServer
Specifies the IP-address of the Time Server in dot-format.

biboAdmTimeOffset
Specifies the time in seconds to add/subtract to the re-trieved time. Values between -24 and +24 are assumed to be hours and are appropriately converted to seconds. Note that when time is retrieved from ISDN the offset must be set to zero.

biboAdmTimeProtocol
Specifies the protocol to use to retrieve current time. Re-spective to four methods noted above the following proto-cols are possible.

- `time_udp`   Time Service (RFC 868) via UDP
- `time_tcp`   Time Service (RFC 868) via TCP
- `time_sntp`   SNTP (RFC 1769) via UDP
- `isdn`   ISDN D-Channel (**stack 0 only**)
- `none`   Disable time retrieval altogether

biboAdmTimeUpdate
Specifies the interval in seconds at which current time should be updated/retrieved. As with Time Offset values between -24 and +24 are converted to seconds. For Proto-col=time_udp, time_tcp, or time_sntp new requests are sent every *biboAdmTimeUpdate* seconds. When isdn is used

the current time will be retrieved from the next ISDN connection established after *biboAdmTimeUpdate* seconds.

Configuration File Loading

• The mechanism used for loading configuration files has been optimized further in Revision 4 and allows for quicker loading of configurations with 100+ partners.

Recommendation: Unless required locally, sites with configurations consisting of 200 or more partners may prefer the following settings in the **admin** table.

```
•biboAdmTrapBrdCast=off
•biboAdmBridgeEnable=disabled
•biboAdmRipUdpPort=0
```

HTTP Enhancements

• HTML pages generated by the htmlshow program now provides buttons for Orientation and Refresh Time automatically.

isdnHistoryMaxEntries

• The *isdnHistoryMaxEntries* variable has been added to the **isdn** table and limits the number of ISDN calls the BRICK saves in the **isdnCallHistoryTable**. Up to 255 calls can be saved using MaxEntries, by default the last 20 calls are saved.

New *debug* Option

• A new timestamp (-t) option has been added to the SNMP shell "debug" command. When the -t option is used each debugging message is preceded by a timestamp:

```
debug -t ether
02:56:34 WARNING/ETHER: Excessive Deferral ....
```

bricktrace

• The bricktrace program (component of BRICKtools for UNIX) now supports CCP (Compression Control Protocol). Using the -p option CCP information can now be displayed.

Bugfixes:

Accounting (ISDN and IP)

- The Date Field in ISDN and IP Accounting messages, was left empty, when the time wasn't set (i.e., date = 1.1.1970).

CAPI

- If the size of the incoming CAPI_DATAB3_IND block exceeds the registered b3 datablock length then the block length is decreased to the maximum registered length.

Configd

- After detecting a syntax error in a configuration file loaded via the TFTP "get" command, the BRICK terminates the transfer and displays the line number where the error was detected.

  ```
  tftp: wrong line <line #>in file <filename>
  ```
  In previous releases the line number displayed was incorrect and hindered subsequent TFTP get requests. This has been corrected in Revision 4.

IPX

- Windows 95's IPX Dial-In-Client uses a different IPX Node Number every time it reconnects. BRICK IPX created new ipxClientTable entries when a Dial-In-Client with a new IPX Node appeared.

  When too many Windows 95 Dial-In-Client reconnections were established the BRICK's *ipxClientTable* overflowed causing a system panic. IPX on the BRICK has been changed to compensate for this.

ISDN

- The remote telephone number wasn't sent in ISDN call_request packets for National ISDN 1. The number is now sent.

PPP

- Unsuccessful calls to ISDN Partner's configured for callback (see: [WAN][EDIT][ADVANCED] Callback) are now correctly attempted *biboPPPMaxRetries* times.

- Charging information stored in the ***biboPPPConnUnits*** and ***biboPPPTotalUnits*** fields were'nt always updated in previous releases and has been fixed in revision 4.

RFC 1086 Support

- A few problems have been corrected relating to the re-implemented (see: Release 4.3 Rev 11) RFC 1086 support on the BRICK. RFC 1086 support now works as described.

RADIUS

- CHAP authentication didn't work properly when used in connection with RADIUS. Revision 4 corrects this.

SNMP

- SNMP traps generated for variables in static tables (i.e., admin, isdn, and system) didn't contain the proper data. This has been corrected.

X.25 and CM-1EBRI

- Routing heavy traffic loads over the ISDN D-channel (i.e. X.31) in conjunction with the CM-1EBRI module sometimes resulted in the local switching station dropping the connection. This has been fixed in Revision 4.

4.4 Revision 2:   Released: 08.04.97

Features:

STAC Compression

The BRICK-XM now supports STAC compression according to RFC 1974 and 1962 (*PPP Stac LZS Compression Protocol* and *PPP Compression Cotrol Protocol* respectively) standards which, depending on the data can increase performance up to a factor of 4. The Stacker LZS algorithm is developed by Hi/fn Inc.

STAC compression on the BRICK is also compatible with Cisco's proprietary STAC implementation which is automatically detected at connection time.

Due to heavy system requirements made by this algorithm only 4 instances can be used simultaneously (i.e. 4 partner connections @ 1 B channel each, or 2 partner connections @ 2 B-channels each).

Note: The software release 4.4 image available at BinTec's FTP server includes STAC support but requires a separate license (purchased separately) to be installed under Setup Tool's LICENSES ➤ menu before it can be used.

HTTPD Server

An HTTP server has been implemented on the BRICK. Currently this daemon provides a status page which can be accessed from WWW browsers supporting HTML tables (RFC 1942) and the HTML 2.0 standard. From the status page you can see which licenses are installed and LAN and ISDN channels that are in use or are available.

Simply point a WWW browser at the BRICK using a URL of the following format (HTTP port number is 80 by default).

http://<*System Name*>**<:***HTTP Port Number*>

The BRICK's http server provides the following features.

SNMP-Table Browsing

The contents of the BRICK's SNMP tables can be browsed via HTTP browsers using the "SNMP Tables" link from the BRICK's main Status-Page. Initially this link displays a list of all system tables found on the BRICK. From there, individual system tables can be selected; the BRICK creates the appropriate HTML pages on-the-fly.

CGI Program: htmlshow

The contents of BRICK SNMP tables and variables can also be selectively displayed to any WWW browser using the htmlshow program. The BRICK authenticates htmlshow queries using the SNMP community passwords (admin, read, write) once per browser session.

The syntax for using htmlshow adheres to the CGI (Common Gateway Interface) standard and can be referenced as follows:

separates CGI program name from parameters

http://<*SysName*>**/htmlshow?**<*option=val*>**&**<*option=val*>

separates parameter strings

where possible options may include:

oid=*snmp_oid*
This option is mandatory and specifies an SNMP object identifier (OID) to display. *snmp_oid* is not case-sensitive. An OID may be specified in one of the following ways:

1. A symbolic object identifier, i.e.
   .iso.org.dod.internet.mgmt.mib-2.interfaces.ifEntry.ifTable

2. An numerical object identifier, i.e.
   .1.3.6.1.2.1.2.2.1

3. A unique MIB-2 or BinTec MIB table or variable name, i.e.
   iftable

Object identifiers starting with a period (".") are taken to be absolute object identifiers; otherwise a relative object identifier is assumed. Relative object identifiers are searched

for relative to MIB-2, i.e. .iso.org.dod.internet.mgmt.mib-2 or .1.3.6.1.2.1.

refreshtime=*interval*
If interval is specified the display is updated every *interval* seconds. Entering 0 in the resulting text field disables automatic refresh updates.

orientation=*mode*
Defines the orientation of the output.
"portrait" (default) or "landscape" mode may be specified.

If more than one object identifier is specified, the resulting tables or columns are printed side-by-side. For example, the following URL was used to display the selected system variables shown below:

http://mybrick/htmlshow?oid=isdnchisdnifinex&
oid=isdnchstate&oid=isdnchreceivedoctets&
oid=isdnchtransmitoctets&oid=isdnchreceivederrors
&refreshtime=10
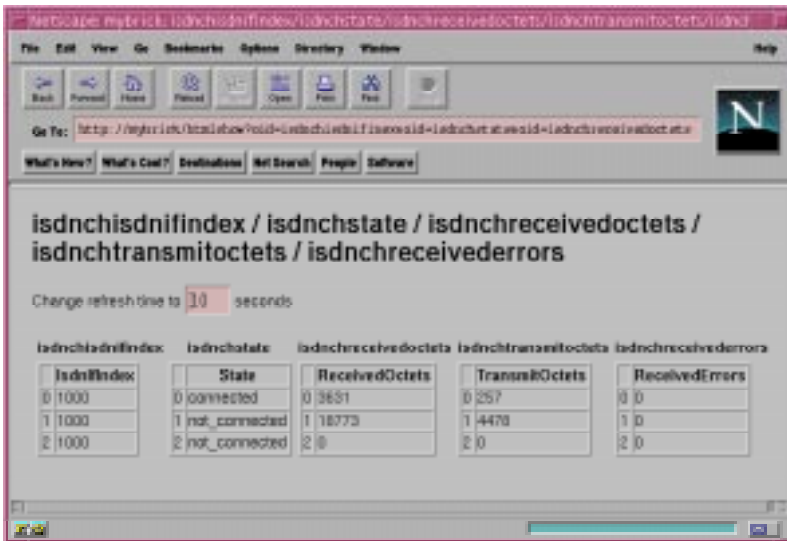
TIP: References to HTML pages generated by the BRICK's htmlshow program can be "bookmarked" for future reference. This will spare you the time of having to type long htmlshow queries (all htmlshow options will be saved in the bookmark, except for SNMP passwords of course).

Path MTU Discovery

- The BRICK supports RFC 1191 which describes a mechanism for discovering the maximum transmission unit (MTU) of an arbitrary internet path.

Enhancements:

CAPI

- Names used in the CAPI Message parameters (when long output is generated) now adhere to the CAPI 1.1 and 2.0 standards. See the Chapter 7 of the *BRICK-XM User's Guide* for information on the format.

Configuration File Loading

- In previous releases loading large configuration files (more than 100 partners) from Flash was often slow. In release 4.4 the loading procedure has been optimized so that large files can be loaded faster.

IPX

- The BRICK now supports IPX connections to remote routers that don't support IPX over WANs according to RFC 1634. Note that the BRICK reverts to standard values for *ipxCircDelay*, *ipxCircNeighRouterName* and *ipxCircType* since negotiating them is not possible with such routers. These routers may need to be configured manually to ensure that only options understood by both are used.

ISDN Callback "Expected"

- ISDN callback is now supported in BOTH directions and can be configured for selected ISDN partners. Callback allows the BRICK to either "call back" an ISDN partner after receiving an initial call, or to "expect" an incoming call from a partner after placing the initial call.

The callback feature can be configured under Setup Tool's WAN PARTNER ➤ ADVANCED SETTINGS ➤ menu using the Callback field (i.e.: yes, no, expected).

When placing the initial call to partners configured for "expected" callback then BRICK moves the interface to the blocked state. Until the remote side returns the call and the line is established packets may be discarded.

X.25

- A special "#" character has been added to the SrcAddr field of the x25RewriteTable. For incoming calls this character is replaced by the incoming caller's ISDN number. For example, if SrcAddr=88#88, then an incoming call from 777 is rewritten as 8877788. (This assumes the caller's ISDN number is transmitted in the source field of the xsw_connect_ind message).

Bugfixes:

capitrace

- A minor change has been implemented in the capitrace application to avoid segmentation faults/core dumps on UNIX machines and Windows 95 systems when decoding incorrect CAPI messages. Hexadecimal longs are now correctly output under 16-Bit Windows systems.

CAPI

- The BRICK now properly transmits a 'user not responding' cause code to the calling CAPI application when an incoming call is ignored by the receiving application.
- Passive FAX polling using the "Layer 3 Protocol T.30 extended" mode (CAPI 2.0) now works properly.
- SELECT_B_PROTOCOL_REQ messages weren't reliable. After a FAX (or other transparent) connection and a subsequent SELECT_B_PROTOCOL_REQ message, incoming calls couldn't connect over this B-channel.

IP

- ICMP "unreachable protocol" messages were sometimes incorrectly transmitted in previous releases.

- In rare cases, the incorrect IP address was transmitted in the source address field of IP frames. This is fixed.

- V.42bis compression sometimes generated conditions that lead to panics in the decompressor. This situation didn't appear often. V.42bis compression now works reliably.

IPX

- The BRICK now correctly sets the number of ticks (*ipxDestTicks*) for routes more than two hops away. In previous releases this led to frequent dialup connections in certain looped network configurations.

- NETX clients couldn't always find the closest server. This has been fixed. Compared to VLM with Packet Burst and LIP enabled, NETX performs badly over WAN links. Recommendation: Consider upgrading to VLM for clients accessing servers via WAN links.

ISDN

- In certain circumstances the BRICK didn't accept isdn-login or PPP connections. The source of this problem was found and has ben corrected in release 4.4.

- Charging information was incorrectly reported in accounting messages (4.3.10 only). This has been corrected.

RADIUS

- When used as a RADIUS client the BRICK sometimes rebooted unexpectedly. This was corrected.

Setup Tool

- The *ipDenySrcIfIndexMode* field is correctly set to verify/dont_verify when deny entries are added to IP access lists.

- Previously, ip_lapb encapsulation for Leased Line interfaces couldn't be configured under Setup Tool. This has been corrected.

TFTP

- TFTP connections were sometimes lost when large configuration files were loaded via TFTP. The sending of ICMP Source Quench Messages was found to be the cause for this and are no longer sent in response to UDP packets.

X.25

- X.25 Diagnostic and Clear codes are now reported in plain text under Setup Tool's X.25 Monitor.

- Sometimes the system was reported to "hang" when the X.25 link was looped, or the cable wasn't terminated.