CAPI SERVER

September 2000

CAPI SERVER

REFERENCE

1 Using the BinTec router as a CAPI Server

1.1 Background on CAPI

CAPI (Common ISDN Application Programming Interface) is an application programming interface used for developing ISDN applications for various operating systems. These applications utilize a host's connected ISDN interfaces. *RVS-COM (Lite Version) for Windows 95 and Windows NT* which is included with the BinTec router is an example of a suite of CAPI applications.

The CAPI specification defines both the CAPI entity (the server) and the protocol that CAPI applications must use when communicating with this entity. By defining a standardized software interface, CAPI allows applications to access ISDN services in a straightforward way. The CAPI specifications are a result of close cooperation among leading ISDN manufacturers and are set forth in two versions.

- CAPI Version 1.1 dated September 1990
- CAPI Version 2.0 dated February 1994

Normally, CAPI is implemented on a single PC with one ISDN adapter and direct access to the ISDN. The CAPI server and the running CAPI applications communicate directly under the shadow of one operating system. Operating systems supporting CAPI include:

- Windows 3.11
- Windows 95
- Windows NT
- Novell
- UNX platforms (using libcapi+capifax)

1.1.1 Register Connect Release

The basic working process of CAPI can be simplified in three basic steps.

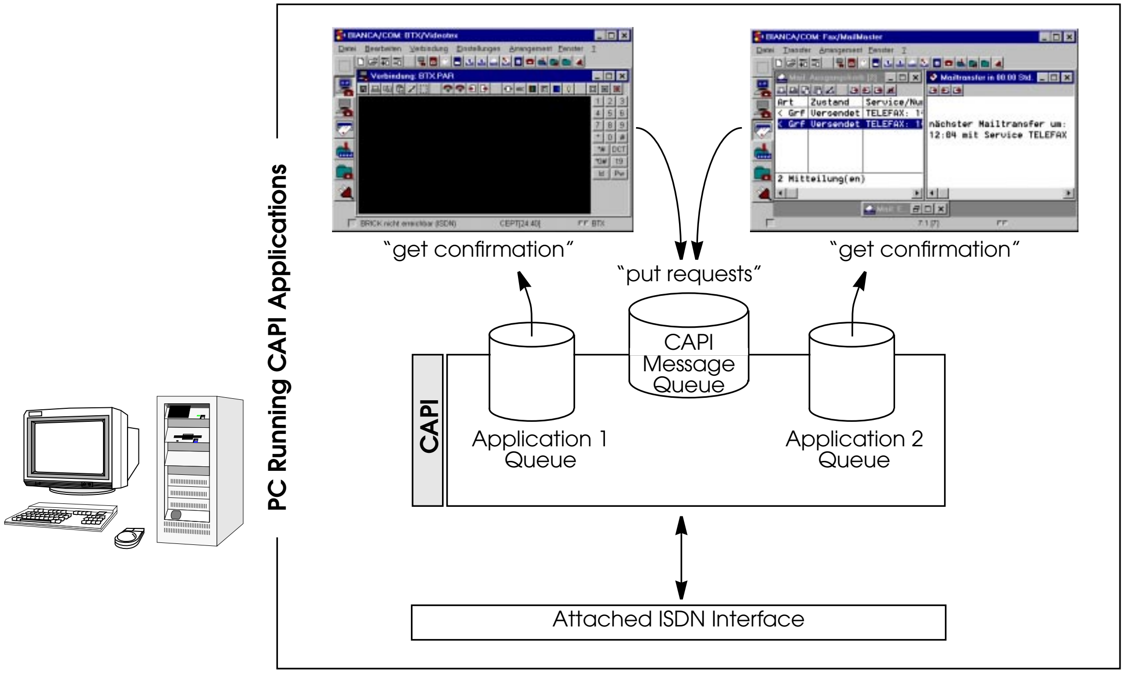
1. **Application Registering**
   Before an application can communicate with the CAPI, it must register with the server. Once it's registered, the server assigns it a unique application ID (APPL_ID). At the same time, the application assigns memory space for its message queue (see below).
2. **Application Connections**
   The application is now ready to establish network connections with other applications using the attached ISDN interfaces.
3. **Application Releasing**
   Just before the application is closed it releases its connection with the CAPI server. In other words the application "un-registers" itself with the server.

1.1.2 Message Queues

CAPI applications communicate with the CAPI server via message queues. A message queue can be seen as a sort of one-way pipe. The CAPI server uses a single message queue to accept messages from all CAPI applications. CAPI applications have their own message queues where they receive responses from the CAPI server.

An application issues commands to an ISDN controller by placing a message in the CAPI's message queue and the

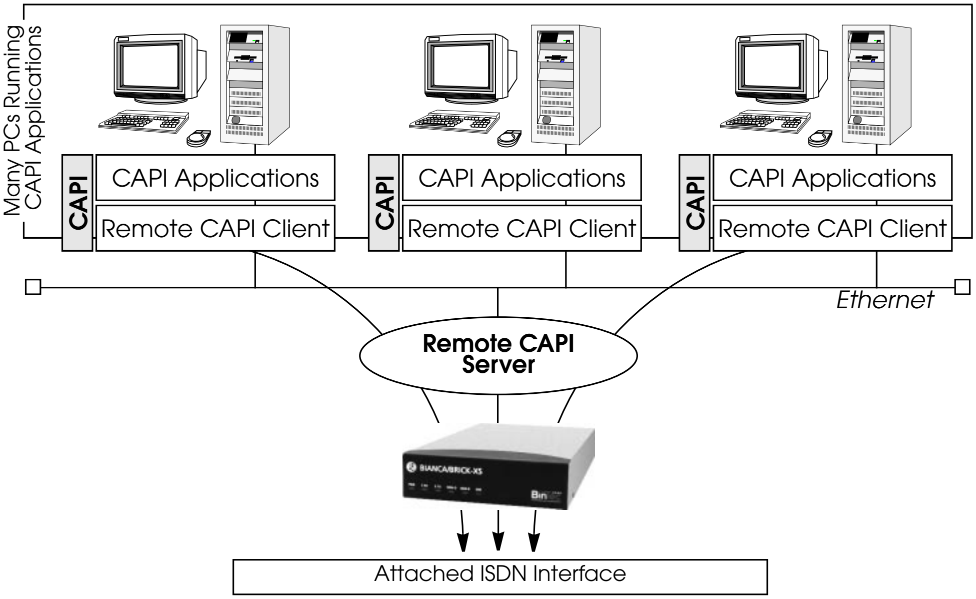
CAPI returns information to the application via its message queue.



As shown above applications put "requests" in the CAPI server's queue and get "confirmation" messages from the server via their respective queues. Sometimes exchanges are initiated by the server. Here, the server puts indication messages in the application's queue and gets a response message from the application via the CAPI message queue.

1.2 The Remote CAPI

Remote CAPI is implemented on the BinTec router and extends the CAPI concept to the network level. As mentioned above CAPI and CAPI applications run on a single PC. Remote CAPI is a client-server system that allows distributed applications running on network computers to access the ISDN interfaces of the BinTec router. A Remote CAPI client is installed on the PC; the BinTec router provides the Remote CAPI Server.



This Remote CAPI is a DualMode server, and supports version 1.1 and 2.0 CAPI. Each PC running Remote CAPI Client software can access the BinTec router's ISDN interfaces as if the interfaces were available locally. To CAPI applications, messaging between client and server remains transparent.

Remote CAPI Library

Using the Remote CAPI server and the accompanying Remote CAPI Library for UNIX, or CAPI.DLL for Windows, existing CAPI applications can be ported to the Remote CAPI environment. The remote CAPI library and header file are included on the Companion CD as **libcapi.c** and **libcapi.h.**

RVS-COM Lite for Windows 95 and Windows NT

In addition to the CAPI library, the BinTec router comes with the CAPI communications package *RVS-COM (Lite Version) for Windows 95 and Windows NT.* RVS-COM Lite consists of several important data communications programs that have been optimized for the Remote CAPI environment.

1.3 CAPI Settings on the BinTec router

1.3.1 CAPI System Tables

The CAPI subsystem on the BinTec router consists of the following system tables. These tables are intended mainly for keeping track of status information and debugging CAPI connections. Most of these variables are related to CAPI internals and will only be of importance to CAPI developers.

```
mybrick: > g capi

73 capiApplTable   74 capiListenTable   75 capiPlciTable             76 capiNcciTable
77 capiInfoTable   78 capiConfigTable   79 capiMultiControllerTable 80 capiUserTable

mybrick: >
```

capiApplTable	Lists all currently connected CAPI applications.
capiListenTable	Contains application-specific listen settings.
capiPlciTable	Contains additional information about

connected applications.

capiNcciTable Contains information for each CAPI NCCI.

capiInfoTable Logs the last 10 CAPI Info Codes and their message identifiers. A list of CAPI Info Codes and their values is contained in Appendix A.

CapiConfigTable Optional settings specific to a local ISDN stack.

capiMultiControllerTable Contains mappings between controller numbers used by CAPI applications and the ISDN stacks available on the BinTec router

capiUserTable Includes authentication settings that control access to the Bintec router's CAPI subsystem.

CapiConfigTable

The ***capiConfigTable*** contains configuration information specific to each ISDN stack on the Bintec router. Each table entry defines settings applicable to all CAPI calls[1] that connect over the respective stack (***capiConfigStkNumber***).

[1] The only exception is the ***capiConfigSendAlerting*** variable.

The default values for the *capiConfigTable* variables are shown below:

```
mybrick: > capiConfigTable

inx StkNumber(*ro)          FaxG3RcvSpeed(rw)      FaxG3ECM(rw)
    FaxG3Header(rw)         VoiceCoding(rw)         SendAlerting(rw)
    V42bis(rw)              ModemDefault(rw)

 00 0                       maximum                off
    logo_header             reverse                voice_only
    off                     modem_profile_1
```

StkNumber As mentioned above, this defines the ISDN stack number the rest of the variables apply to.
Stack numbers are numbered from 0 through 31.

FaxG3RcvSpeed The receive speed to use when receiving G3 faxes.
If a CM-EBRI is connected to this stack or you are using a V!CAS, be sure to set this field to **`maximum`**.

FaxG3ECM Specifies whether error correction mode should be used for G3 facsimile transmissions. Default is on.

FaxG3Header This specifies whether a header-line and/or logo should appear on outgoiing facsimilies. The header--line contains calling information, the logo contains the ***BIANCA/FAX*** symbol.

VoiceCoding Switches the bit order for voice-data.

SendAlerting For CAPI 1.1 this specifies when the CAPI server should send alert messages for incoming calls.

V42bis For V.42bis data compression. Compression is used when ***V42bis***=**`on`** and the remote side supports V.42bis.

capiConfigModemDefault Specifies the modem profile of the ***mdmProfileTable*** which contains the default modem parameters to use for modemconnections. The valid range is from modem_profile_1 to modem_profile_8.

FaxModulation With this variable you can set the following default transmission protocols for fax. The default value is v17.

- – v17 max. 14400 bps new standard
- – v33 max. 14000 bps early standard
- – v29 max. 9600 bps fax standard
- – v17s: v17 with extended fax-on-demand capability
- – v33s: v33 with extended fax-on-demand capability

Fax12000 This enables or disables the 12000bps mode for fax transmission.
If the value of the variable is set to *on*, the fax speed will fall back from 14400 to 12000 bps during a retrain. If set to *off*, it will fall straight back to 9600 bps. The default value is *off*.

FaxTXLevel The transmission level can be set to -x dB db0 = 0dB, db3 = -3dB.
The default value and the value normally used for fax transmission in Germany is -6dB (db6).

capiMultiControllerTable

The *capiMultiControllerTable* was added to the CAPI group to enable the use of CAPI with different ISDN controllers at the same time.
This table contains mappings between controller numbers used by CAPI applications and the ISDN stacks available on the Bintec router (i.e., the ***Number*** field of the ***isdnStkTable***). The ***Version*** field specifies whether an entry applies to a capi11 or capi20 application.
If no CAPI 1.1 entry is defined, CAPI 1.1 applications are assigned ***isdnStkNumber*** n where n is the controller number requested by the application.
If no CAPI 2.0 entry is defined, CAPI 2.0 applications are assigned ***isdnStkNumber*** n-1 where n is the controller number requested by the application.

Creating entries: entries are created by assigning a value to the ***capiControllerNumber*** object.

Deleting entries: an entry can be removed by assigning the value ***delete*** to its ***capiControllerVersion*** object.

The fields of the table have the following meanings:

Number The controller number requested by the CAPI application.

StkMask This binary number defines the ISDN stack(s) to use for the specified CAPI 1.1 or CAPI 2.0 applications. Each bit corresponds to one entry (stack) in the ***isdnStkTable***, the rightmost bit selects entry 0, the next bit selects entry 1, and so forth. For example, ***Number***=1 ***StkMask***=0b1101 ***Version***=capi11 means: allow CAPI 1.1 applications requesting ISDN controller 1 to use ISDN stacks 0, 2 and 3.

Version Specifies which CAPI applications (version 1.1, or 2.0) this entry applies to.
Set this field to ***delete*** to delete this entry.

capiUserTable

The ***capiUserTable*** and ***isdnDispatchTable*** are both parts of the CAPI User Concept.
The CAPI User Concept gives you greater control of access to the Bintec router's CAPI subsystem. Each network user that attempts to access the Bintec router's CAPI subsystem must first be authenticated by using a user name and password which is also configured on the Bintec router, i.e. ***capiUserName*** and ***capiUserPassword*** in the ***capiUserTable*** (it can also be configured over Setup Tool in the ***IP static Settings*** menu). Only if authentication is successful, can the user receive incoming calls or establish outgoing connections via the CAPI.
The fields of the ***capiUsertable*** have the following meanings:

capiUserName The name of the user. Entered on the Bintec router and matches the entry in the Capi Configuration on the BRICKware, as well as the entry in ***isdnDspUserName***.

capiUserPassword The password of the user. Entered on the Bintec router and matches the entry in the Capi Configuration on the BRICKware.

capiUserCapi Allows or denies the use of CAPI.

When an incoming CAPI call arrives at the Bintec router from a WAN partner, the Called Party Number is compared with the ***isdnDspLocalNumber*** in the ***isdnDispatchTable***.

The entire ***isdnDispatchTable*** is only relevant for pure router devices; not for PBX devices.

If the CAPI User Concept is being used, the same user name as configured in ***capiUserName*** is also configured in the ***isdnDspUserName*** in the ***isdnDispatchTable***. The entry in ***isdnDspUserName*** and the entry in ***capiUserName*** are then compared. If they match, the Bintec router then forwards the call only to that CAPI application that is authenticated with the same user name as configured in the ***capiUserTable***.

1.3.2 CAPI TCP Port

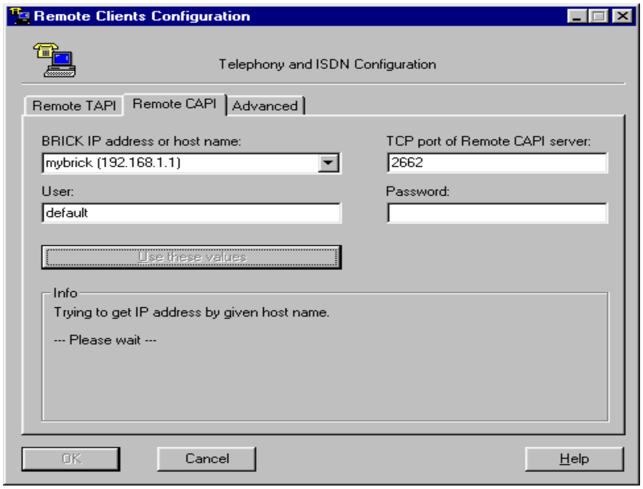
The only required setting on the Bintec router is the CAPI TCP port. This is the port the Bintec router listens to for incoming CAPI connections, i.e. it must match the port number used by the Remote CAPI Client software.

The default port is **2662,** however it can be changed by changing the ***biboAdmCapiTcpPort*** variable.

Note that CAPI can be disabled completely on the Bintec router by assigning port number 0 to the ***biboAdmCapiTcpPort*** variable.

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).



Remote Clients Configuration

Telephony and ISDN Configuration

BRICK IP address or host name: mybrick (192.168.1.1)

TCP port of Remote CAPI server: 2662

User: default

Password:

Info

Trying to get IP address by given host name.

--- Please wait ---

The current Unix Tools “capitrace”, “eft”, and “eftd” 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↵)

1.4 Tracing CAPI Connections

When debugging connections on the Bintec router you may need to trace the ISDN channels to determine why your CAPI connections may be failling. A tracer collects all CAPI messages exchanged between CAPI applications on the LAN and the Bintec router.

Windows 95/Windows NT users will want to use the included *CAPI Trace* program included with *DIME Tools*. *CAPI Tracer* is a CAPI application that communicates directly with the Bintec router via a TCP connection; therefore the installation of the Remote CAPI Client is not required on the trace-host (the host where *CAPI Trace* is started from). For information on using *CAPI Trace*, refer to the *BRICKware for Windows* documentation.

For UNIX systems the **`capitrace`** program is also included on the Companion CD. For information on using the capitrace program, refer to Chapter 7, *Command Reference*, in the *User's Guide*.

1.5 CAPI Features/Enhancements Supported by the Bintec router

The following sections list official enhancements to the CAPI 1.1 and 2.0 standards as well as BinTec specific extensions that are supported by all[1] Bintec router products. As noted earlier, this information will mostly be of interest to CAPI developers.

1.5.1 CAPI 1.1 Enhancements

In addition to the official standards for CAPI Version 1.1 remote CAPI on the Bintec router supports the following enhancements.

- Use of fax group 3
- Support for X.25 PLP on the D-channel
- CAPI-E-DSS1-Mapping
- Management of semi-permanent connections
- Direct Dial In (DDI) for NT1 equipment
- Extension of CAPI error codes/E-DSS1 adaptation
- Support for DTMF[2] functions (receive only)
- NCPD in accordance with ISO 8208 protocol

1.5.2 BinTec Extensions to CAPI 1.1

- **CAPI-DSS1 Mapping** Which maps the Service Indicator and Additional Info (SI and ADD), according to 1TR6, HLC, and LLC.
- **Specification for V.110 Connections** Inband Negotiation is not implemented for synchronous trans-

[1] With exceptions which are appropriately noted.

[2] Dual Tone Multi Frequency is only supported on the V!CAS and products with a CM-1EBRI.

mission using bitrate adaptation according to ITU-T V.110 (user-rate: 10101111).

- **DTMF[2]-Capabilitiy** Only the DTMF-detection (DTMF_IND, and DTMF_RESP messages) and B2-Protocol extensions »0x0B« (i.e. Bittransparent-Transmit Only) are supported.
- ***BinTec specific CAPI-Extensions*** Support for 2400 bps modems (V.22bis).
  The *SELECT_B2_PROTOCOL_REQ* message contains the B2 parameter which is used by modems.

B2	Protocol
0xf0	analog modem, (BinTec-CAPI only) 2400 bps full-duplex: V.22bis

The CAPI-DLPD[1]-Structure for the identification of the »Modem« protocol is coded (analogous to V.110 DLPD) as follows:

DLPD	Meaning
WORD	DATA-LENGTH
BYTE	LINK-ADDRESS-A (not used)
BYTE	LINK-ADDRESS-B (not used)
BYTE	MODULO (not used)
BYTE	WINDOW-SIZE (not used)
BYTE	User Rate (as with V.110)

[1] Data Link Protocol Description

The *Userrate* bitfield relevant for modem operation is coded (analogous to userrate in V.110) as follows:

Userrate Byte	Coding
--1- ----	7 Databits
--0- ----	8 Databits
---- 0---	no parity
---- 1---	even parity

- ***Optional NCPI-Parameters for ISO 8208*** To change the ISO 8208-Default-Windowsize (Standard: 2), these NCPI parameters must be used:

STRUCT NCPI		Default-value
WORD	lic	0
WORD	hic	0
WORD	ltc	1
WORD	htc	1
WORD	loc	0
WORD	hoc	0
BYTE	modulo_mode	8
BYTE	default_window_size (* additional *)	2

- **CAPI 1.1 supported B2/B3 B-channel protocols[1]:**

CAPI 1.1 CAPI_SELECTB2_REQ		BRICK-XM BRICK-XL				BRICK-XS	V!CAS
		CM-1BRI	CM-2BRI	CM-1EBRI	CM-PRI		
0x01	X.75 SLP Basis Operation Mode, with Implementation Rules IAW T.90.	●	●	●	●	●	●
0x02	Transparent-HDLC with Bit-Stuffing, Frame-Detection and CRC-Check	●	●	●	●	●	●
0x03	Bit transparent operation	●	●	●	●	●	●
0x05	X.75 Btx	●	●	●	●	●	●
0x06	Fax G3			●			●
0x07	LAP-D	●	●	●	●	●	●
0x08	V.110 with transparent B2-Protocol			●	●		
0x0a	V.110 with X.75 SLP Basis Operation Mode with Implementation Rules IAW T.90			●	●		
0x0b	Bit transparent operation (transmit only)	●	●	●	●	●	●

[1] DTMF is only supported on the V!CAS and products with a CM-1EBRI.

CAPI 1.1 CAPI_SELECTB3_REQ		BRICK-XM BRICK-XL				BRICK-XS	V!CAS
		CM-1BRI	CM-2BRI	CM-1EBRI	CM-PRI		
0x01	0x01 T.70 NL for circuit switching (CSPDN) (preset).	●	●	●	●	●	●
0x02	ISO 8208 (DTE/DTE).	●	●	●	●	●	●
0x03	Level 3, IAW T.90, Appendix II.	●	●	●	●	●	●
0x04	Transparent.	●	●	●	●	●	●
0x05	Fax T.30.			●			●

1.5.3 CAPI 2.0 Enhancements

The Bintec router supports the CAPI Version 2.0 standard from February 1994 with the following enhancement.

- Support of Direct Dial In (DDI)

1.5.4 BinTec Extensions to CAPI 2.0

- **CAPI 2.0 supported Layer 1, Layer 2, and Layer 3 B-channel protocols[1]:**

CAPI 2.0 Layer 1 Protocols		BRICK-XS	V!CAS	BRICK-XL				
				BRICK-XM				
				CM-1BRI	CM-2BRI	CM-1EBRI	CM-PRI	FML-8MD
0:	64 kBit/s with HDLC framing. This is the default B1 protocol.	●	●	●	●	●	●	
1:	64 kBit/s bit transparent operation with byte framing from the network.	●	●	●	●	●	●	
2:	V.110 asynchronous operation with start/stop byte framing.					●	●	
3:	V.110 synchronous operation with HDLC framing.					●		
4:	T.30 modem for fax group 3.		●			●		●
6:	56 kBit/s bit transparent operation with byte framing from the network.	●	●	●	●	●	●	

[1] DTMF is only supported on the V!CAS and products with a CM-1EBRI.

CAPI 2.0 Layer 2 Protocols		BRICK-XS	V!CAS	BRICK-XL				
				BRICK-XM				FML-8MOD
				CM-1BRI	CM-2BRI	CM-1EBRI	CM-PRI	
0:	ISO 7776 (X.75 SLP). This is the default B2 protocol.	●	●	●	●	●	●	
1:	Transparent.	●	●	●	●	●	●	
3:	LAPD IAW Q.921 for D channel X.25	●	●	●	●	●	●	
4:	T.30 for fax group 3.		●			●		●
5:	Point to Point Protocol (PPP).	●	●	●	●	●	●	
7:	Modem with full negotiation		●			●		●

CAPI 2.0 Layer 3 Protocols		BRICK-XS	V!CAS	BRICK-XL				
				BRICK-XM				FML-8MOD
				CM-1BRI	CM-2BRI	CM-1EBRI	CM-PRI	
0:	Transparent. This is the default B3 protocol.	●	●	●	●	●	●	
1:	T.90NL with compatibility to T.70NL IAW T.90 Appendix II.	●	●	●	●	●	●	
2:	ISO 8208 (X.25 DTE-DTE).	●	●	●	●	●	●	

CAPI 2.0 Layer 3 Protocols		BRICK-XS	V!CAS	BRICK-XL				
				BRICK-XM				FML-8MOD
				CM-1BRI	CM-2BRI	CM-1EBRI	CM-PRI	
3:	X.25 DCE.	●	●	●	●	●	●	
4:	T.30 for fax group 3.		●			●		●
5:	T.30 for fax group 3 (extended)		●			●		
7:	Modem		●			●		●