

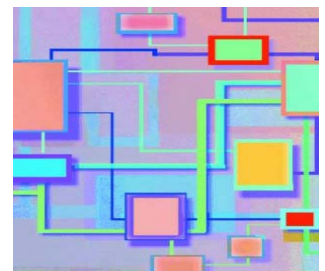
openNet Server (BS2000/OSD) Version 3.2 Communication Manager

Issue September 2007

Pages 6

Communication Manager in BS2000/OSD

“Open networking” is the term used to describe the connection between BS2000/OSD and the different data communication systems. The basic component of open networking in BS2000/OSD is represented by the selectable unit called *openNet Server* (short designation: ONETSERV (BS2000)), which is in turn made up of several products, including BCAM, DCAM, VTSU-B, etc. In the context of *openNet Server* (BS2000), the transport system has been systematically expanded into a communication manager.



Delivery unit

The selectable unit *openNet Server* (BS2000) V3.2 comprises the products BCAM, DCAM, CMX, and SOCKETS (with their specific user program interfaces) and also the products IPSec to the encoded transfer of the user-data and LWRESD for the access on DNS-Server. The user program interfaces enable individual data-communication applications to be implemented.

A standardized trace concept supports the implementation and operation of user programs.

The product VTSU-B is supplied as part of *openNet Server* (BS2000), although it is actually an unbundled, field-replaceable, standalone subsystem.

The additionally product XHCS as part of *openNet Server* (BS2000) is defined as a dynamic subsystem. It is managed by DSSM (Dynamic Subsystem Manager) and loaded during system startup.

BCAM

The product BCAM implements the transport functions in the host computer (end system of a network) within BS2000/OSD open networking.

In addition, a “static routing” procedure is supported for the open system network (ISO, TCP/IP), with BCAM functioning as the “router” between different subnetworks (e.g. LAN, WAN), thus facilitating universal “end-to-end” communication by the partners (“intermediate systems” in a network).

In conjunction with High-speed Net Connect (HNC), fault-tolerant network topologies can be implemented as part of redundancy configurations. Additionally you can separation different user-groups with virtual LANs, called VLAN. BCAM realizes VLAN endpoints with the protocols IEEE 802.1q and 802.1p.

Via the I/O controller, BCAM supports host-channel-attached data communication computers (TRANSDATA 960) running the PDN operating system, the 3612 data exchange controller and the High-speed Net Connect (HNC) for (Fast) Ethernet, Gigabit Ethernet and FDDI connection. On C systems and SX servers, direct network access is provided by means of integrated interface modules.

BCAM supports the protocols of the following architectures:

- NEA - (e.g. NEATT, NEATE)
- ISO - (8073 Cl.0, 2, 4; 9542, 8473)
- TCP/IP - (e.g. UDP, IPv4, IPv6, IPSec)

DNS connection

The names and addresses of the partner systems connected to BS2000/OSD can be stored in external DNS servers for easy and consistent administration. BCAM can access the DNS server, which provides the conversion of names to addresses and vice versa, using the supplied LWRESD product. The DNS server should be redundantly configured for high-availability reasons.

IPSec

Internet technology based on the TCP/IP protocol environment is continuing to develop all the time. For IP, too, there are enhancements aimed at ensuring confidentiality, authentication and integrity based on the use of cryptography. IPSec supports the use of cryptography in layer 3 (network or switching layer) of the OSI reference model.

IPSec provides two protocol-elements for security which can be used independently of each other. The protocol-elements named by its header names and differs in its offered security services. Both protocol-elements are supported by product IPSec.

- The Authentication Header (AH) covers following three security requirements:
 - Authentication of communication partner guarantees that the received package of the right station came.
 - Integrity of the information prevent an unauthorized manipulation, like for example the insertion, leaving out as well as replace of sharings of a message.
 - Measures against replay attacks prevents through the application of sequence numbers that intercepted the data of an invader, duplicates and is transferred again afterwards.

- The Encapsulated Security Payload (ESP) is another protocol-element, that covers following request:
 - Encoding of the information guarantees that no third unauthorized party can reach the content of the message.

According to it, whether one communicates internally in the local network or external over a public network, one has the election between transportation-mode and tunnel-mode.

- **Transport mode:**
The transport mode is used of the majority within a certain internal network. For this reason, the applied security-degree is more inferior than in the tunnel mode. Only the payload of an IP packet is encrypted in the transport mode. Most of the fields of the IP header are not encrypted so that these still remain legible for router.
- **Tunnel mode:**
The tunnel mode is used for connections, which over a public network, like the Internet goes. A higher security degree is usually demanded there. In the tunnel mode, the entire IP packet is encrypted and then packed into the payload part of a further IP packet. The header of the original IP packet is legible through it no longer. The tunnel mode is interesting above all in the context with virtual private networks (VPN).

IPSec uses exclusively symmetrical cryptographically procedures for the guarantee of authenticity, integrity and confidentiality. An additional "sure canal" or a key exchange protocol is required for the key exchange between sender and recipients of a message. This can through a manual exchange or a system with automatically key management takes place. The IPSec implementation in BS2000/OSD supports the automatic key exchange via 'Internet Key Exchange' (IKE) from the correction step 3.2B of *openNet Server*.

DCAM

The product DCAM makes the NEA or ISO transport service available to its user program interface. These services are made possible - using convergence protocols, for example - via all three communication protocol stacks (NEA, ISO, TCP/IP).

DCAM facilitates the following communication relationships:

- user task-to-terminal and
- user task-to-user task.

The communication partners (user task, terminal) may be located in the same or in different processors.

VTSU-B

The product VTSU-B is used for logical support of terminals and terminal printers in "line/page mode". The product FHS may be used in addition for "format mode". To the support of unicode-capable terminal emulations, like for example MT9750 V7.0, VTSU-B was upgraded.

CMX

The product CMX represents the transport access system and has a user program interface (ICMX), as well as using the services of the BCAM transport system. This service is supported partly with the aid of convergence protocols across all three (NEA, ISO, TCP/IP) communication protocol stacks. The ICMX user program interface is also available in the other operating systems (incl. UNIX derivatives, Windows, SINIX and MS-DOS) and provides the ISO Transport Service in the same way as in the BS2000 environment.

Sockets

The product SOCKETS makes the TCP transport service and the runtime environment available for user applications.

XHCS

The XHCS product offers the necessary information on all character sets for all comparison and conversion operations. The products which make up the system therefore no longer

need to maintain the corresponding tables themselves, as previously.

The XHCS interfaces are also available to every user program. The access methods TIAM, DCAM and UTM are linked with XHCS via VTSU. The user can access XHCS with the aid of the corresponding TIAM, DCAM or UTM application programs. UTM users, however, may only access XHCS via FHS. DCAM and TIAM users can call on XHCS services "directly" (via the VTSUCB), or likewise via FHS. XHCS can signs not only between the previous 7-bit- and 8-bit-codes (ASCII, EBCDIC and ISO-8859-x) wanders but also between the codes backed until now and miscellaneous unicode-variations (UTF-8, UTF-16 and UTF-E).

Functional Description

DCAM

The DCAM product provides two forms of the IDCAM interface for implementation of communication applications:

- the NEA transport service and
- the ISO transport service.

The privileged (TPR) BCAM transport services are mapped onto a non-privileged (TU) interface (IDCAM). The ISO transport service is a "pure" transport service based on the OSI reference model. The NEA transport service offers additional functions, e.g.:

- transport acknowledgments
- sequence numbers
- message structuring
- connection password
- long user messages during connection setup.

The IDCAM interface calls can be divided into 4 function groups:

- **Existence function**
Open DCAM application (YOPEN);
Inquire status of a DCAM application (YINQUIRE);
Close DCAM application (YCLOSE).
In addition, for DCAM(NEA) transport service applications, the status of a DCAM application can be changed (YSETLOG).
- **Connection function**
After a DCAM application has been opened, a connection must be set up between the communication partners before data transmission can take place.
The connection setup function comprises the following actions:
 - open connection (YOPNCON),
 - reject request to open connection (YREJLOG);
 - close connection (YCLSCON);
 - change the properties of a connection (YCHANGE).
- **Data transmission function**
After a DCAM application has been opened and a connection set up, data transmission can take place.
The data transmission function comprises the following actions:
 - send a message (YSEND);
 - receive a message (YRECEIVE);
 - send and receive (YSENDREC);
 - cancel receive requests and reset connection status (YRESET).
 For the DCAM(NEA) transport service, the following actions are additionally available for controlling message distribution on the basis of distribution codes:
 - Allocate distribution names to distribution code groups (YPERMIT);
 - Cancel allocation (YFORBID).
- **Name assignment function**
This function enables parameter values for the DCAM application or the connections to be specified at execution time.

For a DCAM application:
the name of the DCAM application;
the password, and, for a DCAM(NEA) transport application,
the distribution name and the password for setting up a
connection.

For the connection:
the name of the partner, the name of the partner's
processor node, and, for DCAM(NEA) transport service
applications, the password for setting up a connection.

These functions enable applications to be implemented with
the following characteristics:

- Logical connection between partners, in which the initiative
for setting up a logical connection may come from the
DCAM application or from the terminal.
- Asynchronous processing facility thanks to asynchronous
execution of certain DCAM macros.
- Event-driven processing by means of special subroutines
that are allocated to specific events, such as arrival of
transport acknowledgments, connection setup requested
by a partner, termination of processing (with the NEA
transport service only).
- Facility for selective message distribution to different user
programs within a DCAM application (via distribution code
forming part of an input message).
- Processing of messages with normal priority and message
telegrams with higher priority.
- Security features to prevent a partner gaining unauthorized
access to a DCAM application (security locks for
applications, codes for users).
- Dynamic name assignment mechanism for assigning
names and passwords which are normally permanently
assigned in the user program; in this way, they do not have
to be generated until execution time (program and
command mode).

Suitable Assembler macros and COBOL calls are available
for implementing applications.

CMX

The CMX product provides the ICMX interface with the ISO
transport service functionality for the implementation of
communication applications. The individual functions fall into
the following categories:

- Attaching and detaching the application
In the attach function, the application transfers its own
address within the local system, its LOCAL NAME, to CMX.
Only then can the application be addressed. On completion
of communication, the application must detach itself from
CMX.
- Setting up a connection
This includes the following functions:
 - Active connection setup
The two functions in this group are used for requesting a
connection to the partner application (connection request)
and for establishing the connection on receiving a
positive response from the partner application
(connection confirmation).
 - Passive connection acceptance
The two functions in this group are used for receiving a
connection setup request from a partner application
(connection indication) and for responding to this request
(connection response).
- Closing down a connection
The two functions in this group are used for closing down a
connection (disconnection request) and for receiving a
disconnection request (disconnection indication)
- Redirect a connection
Within an application, a connection can be forwarded
(rerouted) to a different task in the same application. The
two functions in this group are used for redirecting a

connection (redirect request) and for receiving a
connection from another task (redirect indication).

- Data interchange
These functions enable data to be interchanged as follows:
 - send normal data (data request) and receive normal data
(data indication);
 - send priority data (expedited data request) and receive
priority data (expedited data indication).
Priority data means small amounts of data which are
given preference and transferred to a communication
partner before the main data stream.
- Flow control
The data flow can be controlled separately for normal and
priority data (datastop, datago, xdatastop, xdatgo)
- Retrieving information
This group of functions can be used to obtain information
as follows:
 - Await or retrieve an event, e.g. the closing down of a
connection by the communication partner.
 - Query error.
 - Retrieve information (info) on CMX parameters.
 - Get LOCAL and GLOBAL names, TRANSPORT
ADDRESSES (get local name, get name, get address).
- Synchronizing different events
This function enables a task (the home task or a different
one) to be wakened from the waiting state (wake).

Calls are available in the C programming language for
implementing applications. The CMX program interface is a
library interface.

Sockets

The most important functions to be performed by this
transport user program interface, which features connection-
oriented and connectionless functions, are as follows:

- Setting up a connection between two end points over the
network. Several stages are required by Sockets for setting
up a connection between two end points.
In the first stage, the respective data structure of each of
the end points is defined. The data structure for the end
point addresses is determined in each case by the domain
which creates the socket. A domain is a number of sockets
that share communication attributes, such as name
assignment and address formats. Thus, in order to
establish a transport connection between two points, the
end points are defined. This function supplies a domain
specifier, the socket type (e.g. a connection-oriented
transport connection or a pure datagram service), and a
protocol type. The socket function then supplies a file
descriptor for this socket.
In the second stage, an address is assigned to the end
point in question, i.e. a specific network address is
allocated to it. In the third stage of establishing a transport
connection, a connection request is transmitted to the
communication partner.
In the final stage of a successful connection setup, the
connection request is accepted by the partner.
- Sending and receiving data via the transport connection.
Sockets can operate on a connection-free (Datagram) or
connection-oriented basis.
- Closing down a connection between two end points. The
final function of a transport connection is successful
connection closedown. In Sockets, the "close" function is
used for this. The "close" function closes down the
connection in a controlled way, i.e. all data still awaiting
transmission is sent before the connection is closed down.

The DNS (Domain Name Service) Resolver functionality can
be used in conjunction with the product LWRES.D.

The DNS Resolver function is made possible by the two
Sockets functions 'get host by name' and 'get host by addr'.

Internet Protocol IPv6, the key internet protocol for the future, and also the Internet Control Message Protocol ICMPv6 are implemented as part of the IPv6 suite. The implemented sockets interface permits communication according to IPv6 conventions (RFC 2553). When Sockets is used with addressing according to IPv6 conventions, the Domain Name Service (DNS) is used if required.

The SOCKETS-BS2000-interface V1.0 until V1.3 still became for compatibility will deliver with *openNet Server* V3.2 last. From the next version, SOCKETS-BS2000 V2.x is delivered with the functions described above only more.

BCAM

The BCAM functions include route control, data transfer, buffering of messages, flow control, protocol handling, use of communication processors (WAN connection), channel adapters (LAN connection), and data interchange controllers. The CTC (channel-to-channel) function enables data to be exchanged directly via OCL (Optical Channel Link) "ESCON". BCAM has no user program interfaces. The BCAM services are offered to users of DCAM/CMX/SOCKETS/POSIX-SOCKETS via the user interfaces of those products. Other subsystems, such as TIAM, UTM etc., also make use of BCAM services.

In order to provide an SNMP agent for network, system, and application management in BS2000/OSD, a subagent is made available by BCAM. With the SNMP products in BS2000, read and write access to the objects defined in the MIB II (TCP/IP environment) is supported as standard.

A special BCAM MIB has been implemented in order to enable the transport system to be represented and managed in its totality (NEA, ISO and TCP/IP protocol stack). This MIB is supported by the additional BCAM subagent and the associated BCAM Monitor management application. This monitor can be integrated into a management platform or run as a standalone application.

Because BCAM can access DNS servers via the supplied DNS Resolver and finds the information relating to its partner systems in the network on them, maintaining a separate processor file becomes unnecessary.

VTSU-B

The product VTSU-B (Virtual Terminal Support) permits application programming to be carried out irrespective of differences in the physical characteristics of terminals.

VTSU-B supports the terminal types "LINE TERMINAL" and "PAGE TERMINAL".

The VTSU-B service is available to the user via the user interfaces of the communication access methods.

The integrated "FORMAT TERMINAL" is also available via the COBOL-CALL interface for screen formatting tasks. The software product FHS is additionally required for this.

The P keys of data display terminals are supported with the PLUS utility routine which is part of VTSU-B.

XHCS

The product XHCS is the central source of information on all CCS Coded Character Sets available in BS2000. The implemented functions permit different character sets, and make mechanisms available to all character processing components to enable them to recognize and interpret current character sets. XHCS identifies the data codes regardless of whether they originate from a terminal input, a program output, or from another system. The CCSN Coded Character Set Name is used for identifying the transferred data codes. The data terminals communicate their current character set to the system via an extended terminal protocol and are sometimes able to change this setting dynamically on request. The supported codes are logically assembled into groups of compatible codes according to the character set they contain. Conversions are possible only between codes in the same

group, since XHCS does not recognize corresponding characters in a different group. During data interchange with partner systems the system-wide communication products (Emulation, File Transfer) perform recoding operations at the system boundaries to enable processing to continue seamlessly in the target system.

XHCS supplies the coded character sets in the form of tables. Depending on the requirements of the application, existing character sets can be adapted to local requirements and special character sets added to those already available.

XHCS offers program interfaces for the following functions:

- Provision of various tables of a predefined code (conversion to a different code, conversion from lower to upper case, tables of sorting priorities, and tables of character features).
- Direct conversion of character strings.
- Supplying information on the codes existing in the system and the options for conversion.

These interfaces enable applications to be operated irrespective of the code available, which means that the remaining BS2000 components (e.g. EDT) no longer need to maintain the appropriate tables themselves. This also provides maximum flexibility with regard to additional or user-modified code tables. The XHCS interfaces are also available to all user programs.

Program Description

openNet Server (BS2000) is supplied with the products BCAM,IPSec, LWSERD, DCAM, SOCKETS, CMX, VTSU-B, and XHCS.

BCAM, the "transport system", offers secure, transparent, unstructured duplex transmission of data between freely addressable partners. The IBCAM interface is only accessible for privileged programs, i.e. system programs (TIAM, UTM etc.).

The application layer products, with their user-oriented functions, are based on IBCAM:

- DCAM for inquiry-and-transaction mode or for program-to-program communication via the IDCAM user program interface. Suitable assembler macros and COBOL calls are available for this purpose.
- CMX for program-to-program communication with the ICMX user program interface; this is a library interface and supports applications written in the C programming language. These applications cannot call on the services of VTSU-B.
- Sockets for program-to-program communication with the Sockets user program interface. This interface is also a library interface and supports applications written in the C programming languagecode. The services of VTSU-B cannot be called upon.

With the aid of VTSU-B it is possible to design an application irrespective of differences in the physical characteristics of terminals.

The VTSU-B services are available to the user via the user interface of the various communication access methods.

The XHCS product itself consists almost exclusively of tables, whilst the code to be executed is contained in VTSU-B.

The code tables are set up and modified by means of macros. Both the operating system and the user programs can access these tables.

XHCS is available for privileged and for non-privileged programs.

The interfaces between system applications and XHCS are usually contained in the system-related applications (SORT; IFG; RSO; EDT LMS etc.).

XHCS is defined as a dynamic subsystem. It is managed by DSSM (Dynamic Subsystem Manager) and loaded at system startup, after which it cannot be unloaded.

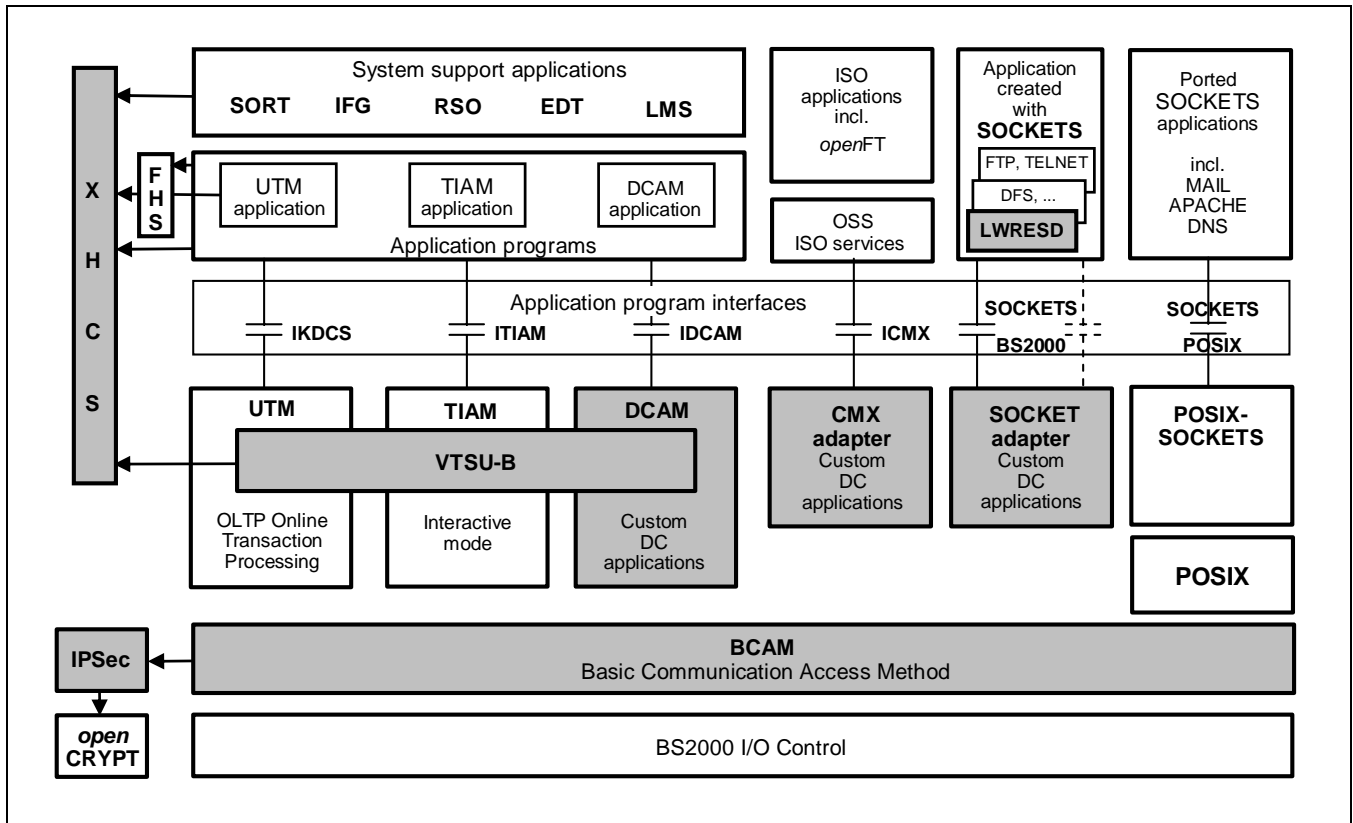


Figure: openNet Server (BS2000) in the system environment (gray denoted)

Technical data

Hardware

openNet Server V3.2 can run on all servers for which a BS2000 release has been approved. This can be seen from the BS2000 classification matrix.

- HNC-III (91851) with software release ≥ V3.09 .
- HNC-IIIIR (91852) with software release ≥ V4.09B .
- HNC-IV (91853) .

openCRYPT™-product are required in order to use cryptographic functions provided when IPSec is deployed.

As of OSD V4.0, TYP-1 channels are no longer supported. This removes the option of connecting to MSN and hence, implicitly, the need for BCAM-LTS. BCAM-LTS is no longer included as part of the product shipment.

The terminals and terminal printers listed below are supported by VTSU:

Terminals:

- 8162 1)
- 8160
- 9748, 9750
- 9751 1)
- 9752 1)
- 9753 1)
- 9749
- 9755, 9756 7), 9758 7), 9762
- 9759, 9763, PC with emulation MT9750, SINIX EM9750
- 3270 6)

Terminal printers*):

- 8121 3)
- 9025
- 8122 3)
- 9003 5)

- 9002 2,3)
- 9001 4)
- 9004 4)
- 9012
- 9013 5), 9014
- 9022
- 9011, 4011,4813
- 3287
- 9021 HP LJII/III, 4810-4812, 4815, 4819-4821, 4824
- 9014

Other devices:

(do not support new functions!)

- Terminals: 8150, 8151, 8152,
- Hardcopy terminals: 8103, PT 80

Note:

The above list includes currently relevant or comparable terminal types.

The terminals/terminal printers shown are sorted according to device type.

- 1) May also be generated as 8160 or 9750
- 2) May also be generated as 8122.
- 3) When printer is connected to 8112:
Only 1 printer per 8112.
- 4) 9001 and 9004 only via the SS97 interface of the terminal.
- 5) No bypass via 8170 or IDS.
- 6) 3270 system: 3278, 3279 and 3180 to SNA-capable controllers 3274 or 3276, with 328x printer as local hardcopy terminal
- 7) Within the scope of 9755 support

*) The following applies for all printers: printing is activated manually or by program control.

Software

For *openNet Server* generally:

BS2000/OSD V5.0, V6.0 or V7.0 (Sxxx line),
BS2000 OSD/XC V2.0 or higher (SX line).

By utilization of the unicode-support in VTSU-B and XHCS:

BS2000/OSD V6.0B with correction package II/2006 or higher
or V7.0 or higher (Sxxx line),
BS2000 OSD/XC V3.0 or higher (SX line).

Further products:

TIAM 13.1 or higher,
PDN-GA V9.0 or higher.

The products PDN-GA V11.0 and NTAC2 V6.0 are required
for the NEA address space extension.

The product SNMP-Basic-Agent BS2000 V3.1 or higher is a
requirement for the SNMP connection to an enterprise
management platform.

The SOCKETS-BS2000-interface V1.0 until 1.3 is delivered
last. From the next *openNet Server*-version, the interface is
delivered from V2.0 only more.

When XHCS is used, the terminals, terminal printers and
terminal emulations listed below are supported:

Terminals:

9756 National (European, Arabic, Persian)
9758 M486
9759 M2/M4
9763 M/C/G/D7
9763 Unicode
EMDS V4.0 Emulation (UNIX derivatives)

Terminal printers:

9001-32
9011-28/29
9012
9013-31x
9014
9021
4819/20

Terminal emulations:

for SINIX: EMDS V4.0 or higher
for MS-Windows: MT9750 V4.0B or higher resp. V7.0 or
higher for unicode

Memory requirement:

The product requires 10 Kbytes of memory and 1,296 bytes
for each code table.

XHCS-capable system support products:

LMSCONV V01.0B or higher;
PLAM V02.0A or higher;
SDF V03.0A or higher;
SDF-U V03.0A or higher;
SHOW-FILE V11.0A or higher;
SYSFILE V11.0A or higher;
ARCHIVE V02.8A or higher;
DPRINT V01.0A or higher;
EDT V16.4A or higher;
HSMS V01.3A or higher;
LMS V02.0B or higher;
OMNIS V06.3A or higher;
OMNIS-MENU V01.3A or higher;
PERCON V02.5A or higher;

RFA V11.0A or higher;
RSO V02.2A or higher;
SDF-A V03.0A or higher;
SORT V07.4A or higher;
TLS V01.0A or higher

Operating mode

Inquiry and transaction mode, interactive mode

Implementation language

BCAM; DCAM; VTSU-B
Assembler macros and COBOL calls
Sockets-DE, CMX
Assembler macros and C calls
XHCS
Assembler, C and SPL

User interface

English/German

Installation

By the user in accordance with the Release Notice.

Documentation

BCAM - User Guide
Volume 1 and Volume 2 (2 manuals).
DCAM - Program Interfaces, Description.
DCAM - Macro Calls,
User Guide.
DCAM - COBOL Calls,
User Guide.
CMX (BS2000)
Communication System in BS2000
VTSU-B
Virtual Terminal Support
Sockets-DE
XHCS
Extended Host Code Support
SNMP User Guide
IPv6 Migration Guide
IPSec – Internet-Security in BS2000/OSD
The documentation is available as online manuals, see
<http://manuals.fujitsu-siemens.com/mainframes.html>, or in
printed form which must be paid for and ordered separately at
<http://FSC-manualshop.com>.

Demands on the user

Knowledge of BS2000/OSD

Training

See course offer at:
<http://www.fujitsu-siemens.com/training>

Conditions

This software product is supplied to the customer under the
conditions for the use of software products against a single
payment or installments.

Ordering and delivery

This software product may be obtained from your local Fujitsu
Siemens Computers GmbH regional office.