

1 Preface

DCAM (Data Communication Access Method) is an access method in the data communication system for program-to-program and program-to-terminal communication. DCAM offers two groups of functions:

- DCAM(NEA) transport service functions
- DCAM(ISO) transport service functions
These offer the user the option of data communication based on the transport services defined by the International Organization for Standardization (ISO).

This User Guide is aimed at programmers wishing to use COBOL language elements in their programs. You should therefore be familiar with this language.

To understand this manual you also require a basic knowledge of BS2000 to enable you to compile and test your programs.

You should also be familiar with the contents of the manual "DCAM Program Interfaces" and the ISO Reference Model.

1.1 The DCAM manual concept

The description of the DCAM COBOL interface is covered by two manuals. The aim is to meet the needs of all users, whether they are programmers or merely want to know more about the subject (see illustration below).

DCAM Program Interfaces

Preface
Basic terminology and introduction to the use of the DCAM interface
DCAM functions
Support for virtual terminals
Coding and executing DCAM programs
Appendix

DCAM COBOL Calls

Preface
Characteristics of the DCAM COBOL interface
Using DCAM functions
Partner characteristics
Examples
Appendix

Structure of the Reference Manual and User Guide

This manual contains the descriptions for both DCAM(ISO) transport service applications and DCAM(NEA) transport service applications. Differences between the two are discussed where applicable. Passages, sections and entire chapters that apply only to DCAM(NEA) transport service applications are indicated by a



at the start of the text.

How to proceed

You should start by carefully studying the "DCAM Program Interfaces" manual. Once you have done this, you will have the necessary background knowledge for the User Guide. The "DCAM Program Interfaces" manual is designed to be read "sequentially". Its aim is to introduce the user to DCAM step by step.

The chapter entitled "**Characteristics of the DCAM COBOL interface**" carries on from the preface of the "DCAM Program Interfaces" manual. It describes the data structures and how they are to be set up in the WORKING-STORAGE SECTION, and also outlines the formal criteria for the calls in the PROCEDURE DIVISION. The section therefore provides a definition of the program framework.

The chapter entitled "**Using the DCAM functions**" contains a detailed description of each call together with the necessary parameter input. It also outlines the feedback information to be evaluated once a call has been executed. The chapter is arranged by function: the individual functions are described in the sequence in which they occur in the program. The structure of the chapter corresponds to that of the chapter on "DCAM functions" in the "DCAM Program Interfaces" manual, so that the latter can be consulted without the need for special cross-referencing.

The chapter entitled "**Partner characteristics**" describes the status information on the partner characteristics.

The chapter entitled "**Examples**" contains a number of examples for DCAM(ISO) transport service applications and DCAM(NEA) transport service applications.

The "**Appendix**" contains all the important lists and tables.

At the back of this User Guide you will find a glossary, a table of references, and a glossary.

A number of books and guides on computer networks and remote data processing with BS2000 deal with topics related to those discussed in this User Guide. Subjects such as generation and administration, programming communication processors and terminals, and support for virtual terminals are dealt with in separate manuals.

1.2 Changes since the last version of the manual

VTSU interface

This manual no longer describes the VTSU interface. Of the chapter that previously bore this name, only the section entitled "Partner characteristics" has been retained. This is because DCAM still supplies the information on partner characteristics for DCAM COBOL applications. See the "VTSU User Guide" for a detailed description of the VTSU interface, the VTSU control block, the logical control characters and the status information.

Feedback information

The table has been expanded to include new feedback information messages.

Readme file

Information on any functional changes and additions to the current product version can be found in the product-specific README file. You will find the file on your BS2000 computer under the name `SYSDOC.product.version.READ-ME.D`. The user ID under which the README file is cataloged can be obtained from your system administrator. You can view the README file using the `/SHOW-FILE` command or an editor, and print it out on a standard printer using the following command:

```
PRINT-FILE FILE-NAME=filename,LAYOUT-CONTROL=PARAMETERS(CONTROL-CHARACTERS=EBCDIC)
```

Any functional changes and additions to the current product version can be found in the [chapter "Manual supplements"](#).


1.3 Notational conventions

In the interests of uniformity and simplicity, the metasympols used in the following descriptions conform to those already widely known from other BS2000 manuals. They are listed and explained in the table below:

Symbol	Explanation	Examples
UPPERCASE LETTERS	Denote constants which the user must specify in this form. In this manual these will normally be literals, which are set in quotation marks.	"YES"
lowercase letters	Denote variables whose contents may vary. The user must replace these upon input with current values. The input format is determined by the type of PICTURE clause.	partnername
{ }	Alternative entries are enclosed in braces and written under one another.	{ "YES" } { "NO" }
[]	Brackets enclose optional entries that can be omitted.	[password4] [{ "YES" }] [{ "NO" }]
<u>underscored</u>	Default values are underscored. The default is the value assumed by the system if no value is specified by the user.	[{ "YES" }] [{ <u>"NO"</u> }]

Symbol	Explanation	Examples
...	Ellipsis indicates repetition. It shows that the preceding specification may be given more than once.	(vsn,...)
()	An expression used to represent variables is placed in parentheses. This symbol shows the range of values at one glance. Since several characters are needed to define a range, the expression must be properly delimited.	(0 < length < 9)
≤	A relation between two values: The value on the left is less than or equal to the value on the right; the value on the right is greater than or equal to the value on the left.	number ≤ 2047 0 ≤ position
≥	A relation between two values: The value on the left is greater than or equal to the value on the right; the value on the right is less than or equal to the value on the left.	number ≥ 1
< >	As above, but without equality.	0 < length quantity < 9

The following notational conventions also appear in the body of the text:

- Note* precedes particularly useful or important items of information
- boldface** highlights important terms
-  indicates passages that are applicable only to DCAM(NEA) transport service applications

2 Characteristics of the DCAM COBOL interface

2.1 Summary

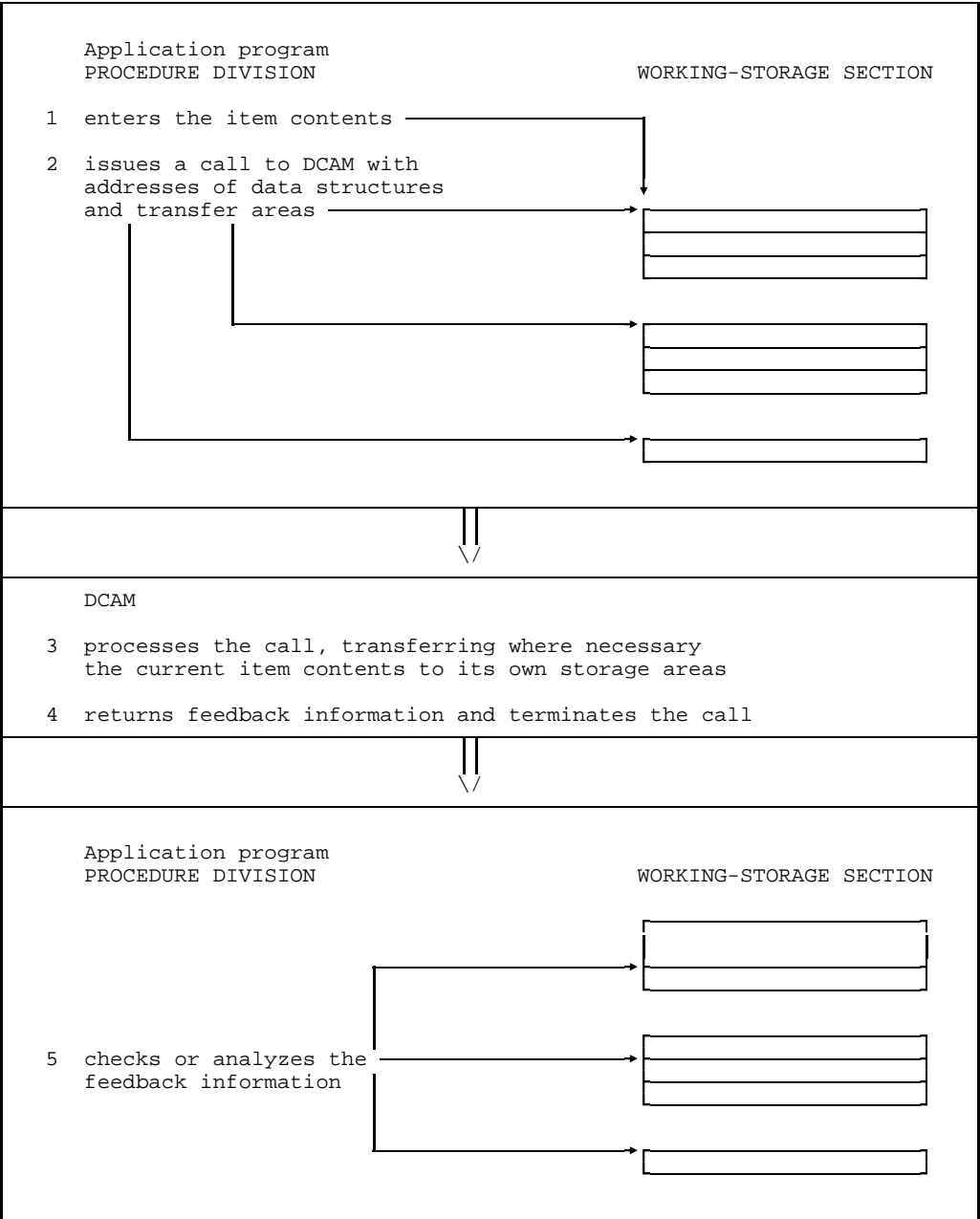
An introduction to the characteristics of the COBOL interface has already been presented in the description of the DCAM program interfaces.

Every call to DCAM requires storage areas for the transfer of data, the so-called data structures and transfer areas. They are provided by the user in the WORKING-STORAGE SECTION.

Every call to DCAM in the instruction portion of the program (PROCEDURE DIVISION) is executed in the following steps:

- Step 1: The data to be transferred to DCAM is entered in the data structures and transfer areas.
- Step 2: The required data structures and transfer areas are specified when the call is issued to DCAM.
- Step 3: In processing the call, DCAM takes the required data out of the transfer areas and stores it in its own storage areas.
- Step 4: DCAM enters feedback information - in any case the return message - into the data structures and transfer areas. Item contents are not otherwise altered. DCAM terminates the call and returns control to the program.
- Step 5: The user analyzes the return information by accessing the items filled by DCAM.

The basic processing steps are summarized in the following diagram.



DCAM COBOL interface

2.2 Description of data structures and transfer areas (WORKING-STORAGE SECTION)

2.2.1 General


Data structures and transfer areas are defined in the WORKING-STORAGE SECTION. They will be described in accordance with the following format:

Level numbers, record names and data names	PICTURE clause	Parameter values to be input for DCAM(ISO) DCAM(NEA)	
--	----------------	---	--

- The **sample entries** for level numbers, record names and data names are suggested values. Users are by and large free to choose their own formats and designations. Those given here correspond to the **descriptive text** and are designed to make it easier to look up the items.
- When stating their PICTURE clause entries, users must adhere to the specifications given in this text. This applies to **type, length** and **sequence** of these items.
- The **parameter values** to be input for the individual items are only **briefly** described in this section. Further information can be found in the functional descriptions (see page 55ff).
- **Default values** need not be specified. They are set whenever an item is blank, i.e. filled with zeros or spaces or if nothing is specified (LOW-VALUE). They are indicated by underlining.
- This section deals only with the entries to be made **by the user**. Entries made by DCAM are only described in the section "Using the DCAM functions".

The following data structures exist:

- application structure
- instruction structure
- connection structure
- wait structure

 and additionally for DCAM(NEA) transport service applications:

- VTSUCB structure (see the chapter entitled "VTSU control block" in the VTSU User Guide)
- distribution structure
- FHS structure

The complete application structure, connection structure and instruction structure must occur in a program **at least once**. The wait structure need only be specified if the program is to wait for the occurrence of an event after asynchronous CALLs.

For DCAM(NEA) transport service applications, the distribution structure is necessary only if message distribution is to take place using distribution codes.

Additional transfer areas are required depending on the call.

Note:

If data structures are to be set up in multiples, the use of the **COPY function** is recommended. It permits multiple duplication of ready-made structures from a COBOL library. Copying is then performed by the COBOL compiler. Copy elements are available for all structures:

YDDCUAPL	for the application structure
YDDCUCOM	for the instruction structure
YDDUCON	for the connection structure
YDDCUWAI	for the wait structure



and additionally for DCAM(NEA) transport service applications:

YDDCUDIS	for the distribution structure
VTSUCBC	for the VTSUCB structure
FHSMAINP	for the FHS structure

As of DCAM V10, the COPY elements (except VTSUCBC and FHSMAINP) are located in the LMS library \$TSOS.SYSLIB.DCAM. The COPY-Element VTSUCBC is located in the LMS library \$TSOS.SYSLIB.VTSU-B.xxx (xxx=VTSU version).

The COPY element FHSMAINP is located in the LMS library \$TSOS.SYSLIB.FHS.xxx (xxx=FHS version).

2.2.2 Application structure (APP-NAME)

Function

A DCAM application is defined in the application structure.

Format

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
01 APP-NAME			
02 APPNAME	PIC X(8).	application-name	
02 PRONAME	PIC X(8).	processor-name	
02 DISNAME	PIC X(8).	-	[distribution-name]
02 LINK	PIC X(8).	[link-name]	
02 LINKMOD	PIC X(3).	{ "PER" } { "TEM" }	
02 LOGPASS	PIC X(4).	LOW-VALUE	[password1]
02 USEPASS	PIC X(4).	[password1]	[password2]
02 USEPW	PIC X(4).	[password2]	[password3]
02 VERIFY	PIC X(3).	{ "PRM" } [{ "SEC" }] { "NO" }	

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
02 ATTR.			
03 SHARE	PIC X(3).		{ "YES" "NO" }
03 LOGON	PIC X(3).	-	{ "YES" "NO" }
03 DISCO	PIC X(3).	-	{ "YES" "NO" }
03 TACK	PIC X(3).	["NO"]	{ "PRI" "REQ" "NO" }
02 FDBK SYNCHRONIZED.			
03 RCD	PIC 9(4) COMP.		
03 ECD	PIC 9(4) COMP.		
03 IND	PIC 9(4) COMP.		
02 ISO	PIC X.		{ "N" "Y" }
02 FILLER	PIC X(8).		

Input

02 **APPNAME** PIC X(8). application-name

application-name is the name of the DCAM application. The first character must be alphabetic (A-Z, @, #, \$). The remaining seven characters can be alphanumeric (A-Z, 0-9, @, \$, #). The name must be entered in this item left-justified, the rest of the item being filled with blanks (SPACES) to the right, if necessary. The use of "\$" as the first character is reserved for system applications.

This item is interpreted for all subroutine calls relating to the specified DCAM application.

02 **PRONAME** PIC X(8). processor-name

processor-name is the symbolic name of the application's own processor.

It is returned for YOPEN.

02 **DISNAME** PIC X(8). distribution-name

distribution-name is the name under which this task can be assigned a distribution code group. The first character must be alphabetic (A-Z, @, #, \$). The remaining seven characters must be alphanumeric (A-Z, 0-9, @, #, \$). The name must be entered left-justified; any remaining positions to the right are filled with blanks (SPACES), if necessary.

This item is interpreted for YOPEN in a primary or secondary task, but only if SHARE = YES and DISCO = YES have been specified in the primary task.

02 **LINK** PIC X(8). link-name

link-name is specified for name assignment (/SET-DCAM-APPLICATION-LINK command, see page 192). The first character must be alphabetic (A-Z, @, #, \$). The remaining seven characters must be alphanumeric (A-Z, 0-9, @, #, \$). The name must be entered left-justified; any remaining positions to the right are filled with blanks (SPACES), if necessary.

This item is interpreted for YOPEN in a primary or secondary task.

```
02 LINKMOD PIC X(3). { "PER"
                      "TEM" }
```

The information from the CLT (communication link table) should be available to the program:

"PER" for the duration of the program run (permanently)

"TEM" for the duration of the call (temporarily).

This item is interpreted for YOPEN in a primary or secondary task, but only if a link name has also been defined.

```
02 LOGPASS PIC X(4). LOW-VALUE/password1
```

LOW-VALUE For DCAM(ISO) transport service applications, LOGPASS must be assigned the value LOW-VALUE.

password1 for DCAM(NEA) transport service applications, specifies a password for establishing a virtual connection. Any partner directing a connection request to this application must also give the password. It may consist of any alphanumeric characters. Blanks (SPACES) mean that no password has been declared.

This item is only interpreted for YOPEN by a primary task and, for DCAM(NEA) applications, only if the LOGON attribute has also been assigned the value "YES".

```
02 USEPASS PIC X(8). password1/password2
```

password1/2 specifies a password for admitting a secondary task to an existing application. Each secondary task must have this password specified in its USEPW operand if it also wants to open the DCAM application concerned. The password may consist of any alphanumeric characters. Blanks (SPACES) mean that no password has been declared.

This item is only interpreted for YOPEN by a primary task if the SHARE attribute has also been assigned the value "YES".

02 USEPW PIC X(4). password2/password3

password2/3 is the password for being admitted to an existing application as specified by the primary task in its USEPASS operand or as specified in the generation of the application. The password may consist of any alphanumeric characters. Blanks (SPACES) mean that no password has been declared.

This item is interpreted for

- YOPEN by a secondary task if the password has been declared in the primary task.
- YOPEN by a primary or secondary task if the application has been protected by a password in the generation.

02 VERIFY PIC X(3). $\left\{ \begin{array}{l} \text{"PRM"} \\ \text{"SEC"} \\ \text{"NO"} \end{array} \right\}$

The task opening the DCAM application wants to be

"PRM" the first one to open the application (PRIMARY), or

"SEC" a subsequent opener (SECONDARY), or

"NO" it does not matter (not checked).

This item is interpreted for YOPEN by a primary or secondary task. If the result of the check is negative, the call will not be executed.

03 SHARE PIC X(3). $\left\{ \begin{array}{l} \text{"YES"} \\ \text{"NO"} \end{array} \right\}$

The DCAM application can be used either by

"YES" multiple tasks (SHARE = shareable), or

"NO" a single task only (NSHARE = non-shareable).

This item is interpreted for

- YOPEN by a primary task,
- YOPEN by a secondary task: "YES" is mandatory for the secondary task, otherwise ("NO" or omitted) the call will not be executed.

```
03 LOGON PIC X(3). { "YES"
                    "NO" }
```

For NEA transport service only: a request to set up a connection is either

"YES" processed (LOGON), or
 "NO" not processed (NLOGON).

This item is interpreted for YOPEN by a primary task.

```
03 DISCO PIC X(3). { "YES"
                    "NO" }
```

For NEA transport service only: message distribution is to take place

"YES" using distribution codes in the message. In this case, SHARE must have been assigned the value "YES".

"NO" as specified by the CS entry in the instruction structure for a YOPNCON, YSEND or YRECEIVE call.

The setting of CS determines the type of queue:

CS="NO": "Continue-Any state" selects the common receiver queue, while

CS="YES": "Continue-Specific state" selects the originator-oriented queue.

Messages are not distributed via distribution codes.

The item is interpreted for YOPEN by a primary task.

```
03 TACK PIC X(3). "NO" / { "PRI"
                          "REQ"
                          "NO" }
```

A message transport acknowledgment is either to be sent

"PRI" (NEA transport service only) to the primary task (PRIMTASK), or

"REQ" (NEA transport service only) to the requesting task (REQTASK), or

"NO" is not to be sent at all (NOTACK).

This holds for the entire duration of the application. Entries in YSEND must take this into account.

This item is interpreted for YOPEN by a primary task.

03 ISO PIC X(3). $\left\{ \begin{array}{l} \text{"N"} \\ \text{"Y"} \end{array} \right\}$

With DCAM(ISO), this item must be assigned the value "Y", and with DCAM(NEA), the value "N".

This item is interpreted for YOPEN by a primary or secondary task.

2.2.3 Instruction structure (BEF-NAME)

Function

The operands for executing a DCAM function are defined in the instruction structure.

Format

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
01 BEF-NAME			
02 TOVAL	PIC 9(4) COMP.	$\left[\begin{array}{l} (0 < n \leq 9999) \\ 600 \end{array} \right\}$	
02 SEQNO	PIC 9(4) COMP.	-	$\left[(0 < n \leq 9999) \right]$
02 OPTCD.			
03 SPEC	PIC X(3).	$\left\{ \begin{array}{l} \text{"YES"} \\ \text{"NO"} \end{array} \right\}$	
03 CS	PIC X(3).	$\left\{ \begin{array}{l} \text{"YES"} \\ \text{"NO"} \end{array} \right\}$	
03 Q	PIC X(3).	$\left\{ \begin{array}{l} \text{"YES"} \\ \text{"NO"} \end{array} \right\}$	
03 ACCPT	PIC X(3).	$\left\{ \begin{array}{l} \text{"YES"} \\ \text{"NO"} \end{array} \right\}$	
03 STP	PIC X(3).	-	$\left\{ \begin{array}{l} \text{"YES"} \\ \text{"NO"} \end{array} \right\}$
03 DG	PIC X(3).	$\left\{ \begin{array}{l} \text{"ELE"} \\ \text{"GRP"} \end{array} \right\}$	$\left\{ \begin{array}{l} \text{"ELE"} \\ \text{"SUB"} \\ \text{"GRP"} \end{array} \right\}$

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
03 NORMAL	PIC X(3).	"YES"	[{"YES"} {"NO"}]
03 TRUNC	PIC X(3).		[{"TRC"} {"KEE"} {"VTK"}]
03 TACK	PIC X(3).	-	[{"YES"} {"NO"}]
03 BELL	PIC X(3).	-	[{"YES"} {"NO"}]
03 FHS	PIC X(3).	-	[{"YES"} {"NO"}]
02 FDB SYNCHRONIZED.			
03 ASEQNO	PIC 9(4) COMP.		
03 TACKNO	PIC 9(4) COMP.		
03 ARECLN	PIC 9(4) COMP.		
03 FDBK.			
04 RCD	PIC 9(4) COMP.		
04 ECD	PIC 9(4) COMP.		
04 IND	PIC 9(4) COMP.		

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
03 REASON REDEFINED FDBK.			
04 LOSCON-REASON.	PIC 9(4) COMP.		
88 USER-DISCON	VALUE 0.		
88 INVALID-EDIT-DEPROT	VALUE 4.		
88 PTN-PROC-ERR	VALUE 8.		
88 PTN-NOT-AVAIL	VALUE 12.		
88 SYSTEM-DISCON	VALUE 16.		
88 PTN-LOST-CONN	VALUE 20.		
88 DISCON-NETWORK	VALUE 24.		
88 DISCON-WARNING	VALUE 32.		
88 PTN-CHAR-NACC	VALUE 36.		
88 ADM-DISCON	VALUE 40.		
88 SERVICE-DATA-ERR	VALUE 44.		
88 SERVICE-PROT-ERR	VALUE 52.		
88 TRANSP-SYSTEM-ERR	VALUE 56.		
88 SYNTAX-ERR-STA	VALUE 64.		
88 NETW-PRIO-NACC	VALUE 84.		
88 PTN-SYSTEM-SHORT	VALUE 88.		
88 PTN-VTSU-ERR	VALUE 92.		
88 PTN-PROT-ERR	VALUE 96.		
88 PERM-ERR	VALUE 100.		
88 WRONG-STAT-NAME	VALUE 104.		
88 WRONG-PROC-NAME	VALUE 108.		
88 NO-TASK-CREATED	VALUE 112.		
88 WRONG-CID	VALUE 116.		

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
88 INT-AUT-ERR	VALUE 120.		
88 ECRNAM-ERR	VALUE 124.		
88 DCAM-VERSION	VALUE 128.		
88 NO-CHIP-CARD	VALUE 132.		
88 PROT-INCON-ERR	VALUE 136.		
88 NO-CHIP-CARD-SS	VALUE 140.		
88 KVP-PROT-ERR	VALUE 144.		
04 FILLER	PIC X(4).		
02 OPTCD1.			
03 SYN	PIC X(3).		{ "YES" "NO" }
03 VTSUCB	PIC X(3).	-	{ "YES" "NO" }
03 FILLER	PIC X(12).		
02 FILLER	PIC X(9).		

Input

02 TOVAL PIC 9(4) COMP. $\left. \begin{matrix} (0 < n \leq 9999) \\ 600 \end{matrix} \right\}$

n is the time a call can be kept waiting when it cannot be executed immediately. The maximum value is 9999 seconds. If the COBRUN parameter 'TRUNC=NO' is used, the maximum value which can be specified is 32767 seconds. Deviations of a few seconds are possible (depending on task management). 600 seconds is the default value.

Exception

TOVAL=ZERO means unlimited wait time

This item is interpreted for

- YOPNCON with ACCPT="YES" and Q="YES",
- YRECEIVE, but only if Q="YES" has also been entered.

02 SEQNO PIC 9(4) COMP. (0 < number ≤ 2048)

For NEA transport service only:

number is the sequence number of the message to be sent. The maximum value is 2048. If the COBRUN parameter "TRUNC=NO" is used, the maximum value which can be specified is 32767.

This item is interpreted for YSEND.

The sequence number is passed on to the communication partner and made available in the ASEQNO item following execution of a YRECEIVE. If an acknowledgment is requested for the message, the number of the message acknowledged is contained in the TACKNO item once the acknowledgment has been received via YRECEIVE.

03 SPEC PIC X(3). { "YES"
"NO" }

"YES" The call relates to a specific partner (SPEC).

"NO" The call relates to any partner (ANY).

This item is interpreted for YOPNCON with ACCPT="YES", YRECEIVE and YRESET.

03 CS PIC X(3). { "YES"
"NO" }

Following execution of the call, all further data transmitted from this partner is to be entered

"YES" in the originator-oriented queue (=CS state of the connection: all further messages arriving from this partner are entered in the originator-oriented queue, from where they can only be picked up with SPEC="YES"). In the case of a shareable DCAM application, only the task that caused the CS state may receive the subsequent messages.

"NO" in the command receiver queue (= CA state of the connection: all further messages arriving from this partner are entered in the common receiver queue, from where they can only be picked up with SPEC="NO").

This item is interpreted for YOPNCON, YSEND, YRECEIVE and YRESET (only with SPEC="YES"); for DCAM(NEA) transport service applications, but only when distribution codes are not in use (DISCO="NO" in the application structure for YOPEN of the primary task);

```
03 Q PIC X(3). { "YES" }
                { "NO" }
```

If the call cannot be executed immediately, then it is to be

"YES" entered in a queue (the maximum wait time is specified in TOVAL). For an asynchronous call the system always enters "YES".

"NO" terminated at once (NQ = no entry in a queue).

The item is interpreted for YOPNCON with ACCPT="YES" and YRECEIVE.

```
03 ACCPT PIC X(3). { "YES" }
                   { "NO" }
```

A request to establish a connection is to be

"YES" accepted (=ACCPT),

"NO" issued (=ACQUIRE).

This item is interpreted for YOPNCON.

```
03 STP PIC X(3). { "YES" }
                 { "NO" }
```

For NEA transport service only: the DCAM application opened by this task is to be placed in the

"YES" STOP state (connection requests are rejected),

"NO" START state (connection requests are processed).

This item is interpreted for YSETLOG, but only if the application has been opened with LOGON="YES".

```
03 DG PIC X(3). { "ELE" } / { "ELE" }
                  { "GRP" }   { "SUB" }
                  { "GRP" }   { "GRP" }
```

The message being sent is

"ELE" an ELEMENT of a subgroup or group of data,

"SUB" the last element of a SUBGROUP of data (NEA transport service only),

"GRP" not subdivided or it is the last element of a GROUP of data.

This item is interpreted for YSEND:

- for DCAM(ISO) transport service applications, only if the connection is using the more-data function. If the connection was established with MDATA="N", the data is always sent with DG="GRP". Note that the way the message is divided up at the receive end and at the send end does not have to be the same (see the "DCAM Program Interfaces" manual).
- for DCAM(NEA) transport service applications, only if the addressed connection has been established with EDIT="USR" or EDIT="SYS" and TRANSF="PHY".

```
03 NORMAL PIC X(3). "YES" / { "YES" }
                              { "NO" }
```

The message to be sent is to be transmitted

"YES" with normal priority; for DCAM(ISO) transport service applications, the NORMAL item must be assigned the value "YES",

"NO" (NEA transport service only) with high priority (= EXPRESS) and, if possible, delivered to its destination at once (in the case of ASSEMBLER programs with asynchronous EXPR notification; see "DCAM Macros").

This item is interpreted for YSEND. For DCAM(NEA) transport service applications, it is only interpreted if the connection has been established with EDIT="USR"; otherwise the message will be rejected with ECD=76 and RCD=12 (see page 177ff).

```
03 TRUNC PIC X(3). { "TRC"  
                   "KEE"  
                   "VTK" }
```

If the message received is longer than provided for in the reception area, the excess portion is to be

"TRC" cut off (message truncated) and discarded (TRUNC),

"KEE" kept and indicated (KEEP), so that the excess portion can be picked up by another YRECEIVE after the CS state has been set (CS="YES"),

"VTK" handled as preset in the connection structure for this connection.

This item is interpreted for YRECEIVE, but only with messages that are longer than the reception area.

```
03 TACK PIC X(3). { "YES"  
                   "NO" }
```

For NEA transport service only: a positive transport acknowledgment for the message sent is

"YES" requested,

"NO" not requested (=NTACK).

This item is interpreted for YSEND; note, however, that the sequence number must also be defined in SEQNO to enable the message to be paired off with its acknowledgment when the latter is received.


```
03 BELL PIC X(3). { "YES" }
                  { "NO" }
```

For NEA transport service only: after an output to the terminal

"YES" an audible alarm is activated,

"NO" an audible alarm is not activated.

The item is interpreted for YSEND, but only if the connection was established with EDIT="SYS".

Note

The BELL function is only possible for those terminals with the appropriate device option (8160, 8161, 8162, 9750 DDTs).

```
03 FHS PIC X(3). { "YES" }
                 { "NO" }
```

For NEA transport service only: messages are input/output

"YES" in formatted mode using the integrated Format Handling System FHS,

"NO" without support from the integrated Format Handling System FHS.

The item is interpreted for YSEND and YRECEIVE.

Note

If messages are to be input/output with FHS="YES", the user must enter the following parameter values, either at connection setup ("YOPNCON") or when using the function "Changing the characteristics of a connection" ("YCHANGE"), in the items below:

"SYS" for message editing by the system	in EDIT,
"FOR" for format handling during input	in EDITIN,
"FOR" for format handling during output	in EDITOUT.

The chargeable software product FHS is required if FHS="YES" is specified. An INCLUDE statement (see description of TSOSLNK) must be used to link the FHS module MFHSCALL to the program. If the module is not made available, the call containing FHS="YES" is rejected with a return code.

```
03 SYN PIC X(3). { "YES " }  
                  { "NO " }
```

The call is to be made

"YES" synchronously,

"NO" asynchronously.

Note

Control is returned to the user immediately after the acceptance processing of the call. Termination of the call is entered in the event queue of the task (FIFO) and the user can retrieve the events from the queue with YWAIT. Up to 8 asynchronous CALLs of the same type or 128 YOPNCON (ACQUIRE) calls can be sent simultaneously.

The call is terminated after successful execution, timeout or after closing of the connection or application.

```
03 VTSUCB PIC X(3). { "YES " }  
                   { "NO " }
```

For NEA transport service only:

"YES" VTSUCB is also to be specified for message editing,

"NO" VTSUCB not present.

This item is interpreted for YSEND and YRECEIVE. VTSUCB is described in the chapter entitled "VTSU control block" in the "VTSU User Guide".

2.2.4 Connection structure (CONN-NAME)

Function

The virtual connection is defined in the connection structure.

Format

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
01 CONN-NAME			
02 PTNNAME	PIC X(8).		[partner-name]
02 PRONAME	PIC X(8).		[processor-name]
02 PASSWORD	PIC X(4).	-	[password4]
02 LINK	PIC X(8).		[link-name]
02 DEPROT	PIC X(3).		
02 LINKMOD	PIC X(3).		{ "PER" "TEM" }
02 EDITIN.			
03 TRANSF	PIC X(3).	-	{ "PHY" "LIN" "FOR" }
03 GETBS	PIC X(3).	-	{ "YES" "NO" }
03 GETFC	PIC X(3).	-	{ "YES" "NO" }
03 LCASE	PIC X(3).	-	{ "YES" "NO" }

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
02 EDITOUT.			
03 TRANSF	PIC X(3).	-	{ "PHY" "LIN" "FOR" }
03 HCOPY	PIC X(3).	-	{ "YES" "NO" }
03 HOM	PIC X(3).	-	{ "YES" "NO" }
02 EDIT	PIC X(3).	-	{ "USR" "SYS" }
02 PROC.			
03 TRUNC	PIC X(3).		{ "YES" "NO" }
03 SYSCODE	PIC X(3).	-	{ "YES" "NO" }
03 APPSTART	PIC X(3).	-	{ "YES" "NO" }
02 MAXLN	PIC 9(4) COMP.		{ n ≤ 9999 4096 }
02 PTNCHAR.			
03 PTNTYPE	PIC 9(4) COMP.		
03 DEVTYPE	PIC 9(4) COMP.		
03 CHARSET	PIC 9(4) COMP.		

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM(ISO)	DCAM(NEA)
03 DEVSEC	PIC 9(4) COMP.		
02 EDITOUT1.			
03 EXTND	PIC X(3).	-	[{"YES"} {"NO"}]
03 LOGC	PIC X(3).	-	[{"YES"} {"NO"}]
03 LACK	PIC X(3).	-	[{"YES"} {"NO"}]
03 FILLER	PIC X(15).		
02 PROC1			
03 SIGNAL	PIC X(3).		[{"YES"} {"NO"}]
03 TERMSTAT	PIC X(3).	-	[{"YES"} {"NO"}]
03 FILLER	PIC X(6).		
02 MDATA	PIC X.	[{"Y"} {"N"}]	-
02 FILLER	PIC X(2).		
02 RLTH	PIC 9(8) COMP.	0≤n≤65535	-
02 ROUTN	PIC 9(4) COMP.		0≤n≤8
02 ROUTL	PIC X(8).		[route-name]
02 FILLER	PIC X(12).		

Input

02 **PTNNAME** PIC X(8). partner-name

partner-name is the name of the communication partner being addressed. The first character must be alphabetic (A-Z, @, #, \$). The remaining seven characters can be alphanumeric (A-Z, 0-9, @, #, \$). The name must be entered in the item leftjustified; any remaining positions to the right are filled with blanks (SPACES), if necessary.

This item is interpreted for

- YOPNCON with ACCT="NO" or with ACCT="YES" and SPEC="YES",
- YINQUIRE with the function "PTN" (=PTNCHAR),
- YINQUIRE with the function "TOP" (=TOPLOGON); here DCAM enters the name,
- YCHANGE
- YCLSCON
- YSEND
- YRECEIVE with SPEC="YES",
- YRECEIVE with SPEC="NO"; here DCAM enters the name,
- YRESET with SPEC="YES".

02 **PRONAME** PIC X(8). processor-name

processor-name is the name of the processor node to which the partner is connected as a station. It is entered together with the partner name and the two entries constitute the full address of the communication partner. It has the same format as the partner name and is interpreted with the latter.

02 **PASSWORD** PIC X(4). password4

password4 is the password for establishing a connection, as prescribed by specifying LOGPASS password1 for APP-NAME in the YOPEN or by using the /SET-DCAM-APPLICATIO-LINK command.

This item is interpreted for YOPNCON with ACCEPT="NO", but only if a password has been defined.

02 LINK PIC X(8). link-name

link-name is specified (LINK) for name assignment (/SET-DCAM-CONNECTION-LINK command, see page 192). The first character must be alphabetic (A-Z, @, #, \$). The remaining seven characters can be alphanumeric (A-Z, 0-9, @, #, \$). The name must be entered left-justified; any remaining positions to the right are filled with blanks (SPACES), if necessary.

This item is interpreted for YOPNCON.

02 LINKMOD PIC X(3). $\left\{ \begin{array}{l} \text{"PER"} \\ \text{"TEM"} \end{array} \right\}$

The information from the CLT (communication link table) is to be available to the program:

"PER" for the duration of the program run (permanently),

"TEM" for the duration of the call (temporarily).

This item is interpreted for YOPNCON, but only if a link name has also been defined.

03 TRANSF PIC X(3). $\left\{ \begin{array}{l} \text{"PHY"} \\ \text{"LIN"} \\ \text{"FOR"} \end{array} \right\}$

For NEA transport service only: when the system edits the messages for input,

"PHY" no virtual (logical) terminal is to be used (= "PHYS"),

"LIN" the line terminal is to be used. The system edits messages with the help of the line terminal. Messages can be structured using logical control characters (see page 50),

"FOR" the format terminal is to be used, i.e. a freely selectable formatting routine (e.g. FORM) or the integrated Format Handling System (FHS) can be used.

This item is interpreted for YOPNCON and YCHANGE, but only if EDIT="SYS" has been specified.

```
03 GETBS PIC X(3). { "YES"
                   "NO" }
```

For NEA transport service only: the backspace characters used for input are to be

"YES" passed to the user,

"NO" interpreted by the system as delete symbols and thus removed together with the characters to be deleted (= NGETBS: backspace characters are not passed on).

This item is interpreted for YOPNCON and YCHANGE, but only if EDIT="SYS" and TRANSF="LIN" or "FOR" are specified.

Note

This function is only meaningful with devices that have a backspace character in their character set, such as the 8103 Printer Terminal.

```
03 GETFC PIC X(3). { "YES"
                   "NO" }
```

For NEA transport service only: the logical function code of a terminal

"YES" is to be the first character of a message received,

"NO" is not to be transmitted.

This item is interpreted for YCHANGE and YOPNCON, but only if EDIT="SYS" and TRANSF="LIN" are specified.

```
03 LCASE PIC X(3). { "YES"
                   "NO" }
```

For NEA transport service only: lowercase letters in the input are

"YES" not converted but rather passed encoded in EBCDIC as lowercase letters,

"NO" converted to uppercase letters (= NLCASE: the DCAM application receives uppercase letters only, even if, for instance, texts containing lowercase letters have been sent from the terminal).

This item is interpreted for YOPNCON and YCHANGE, but only if EDIT="SYS" has been specified.


```
03 TRANSF PIC X(3). { "PHY"
                     "LIN"
                     "FOR" }
```

For NEA transport service only: when the system edits the messages for output,

"PHY" no virtual terminal is to be used (= PHYS),

"LIN" the line terminal is to be used. The system edits messages with the help of the line terminal. Messages can be structured using logical control characters (see page 50),

"FOR" the format terminal is to be used, i.e. a freely selectable formatting routine (e.g. FORM) or the integrated Format Handling System (FHS) can be used.

This item is interpreted for YOPNCON and YCHANGE, but only if EDIT="SYS" is also specified.

```
03 HCOPY PIC X(3). { "YES"
                    "NO" }
```

For NEA transport service only: the message to be output

"YES" is also to be printed out on an available printer (hardcopy device),

"NO" is not to be printed out (=NHCOPY).

This item is interpreted for YOPNCON and YCHANGE, but only if EDIT="SYS" and TRANSF="LIN" have also been specified. With TRANSF="FOR", the operands of the format statements are used for this function.

```
03 HOM PIC X(3). { "YES"
                  "NO" }
```

For NEA transport service only: specifies how logical lines are handled.

"YES" all logical lines of a message are regarded as a unit,

"NO" each individual logical line is regarded as a unit.

This item is interpreted for YOPNCON and YCHANGE, but only if EDIT="SYS" and TRANSF="LIN" are specified.

```
02 EDIT PIC X(3). { "USR"
                  "SYS" }
```

For NEA transport service only: messages are edited by

"USR" the user (= USER: any appropriate method of data transfer between DCAM applications),

"SYS" the data communication system (= SYSTEM: virtual terminals may be used for I/O; messages must be in EBCDI code). EDIT="SYS" is a prerequisite for the use of VTSUCB.

This item is interpreted for YOPNCON.

```
03 TRUNC PIC X(3). { "YES"
                   "NO" }
```

If the message received is longer than provided for in the reception area, the excess portion is to be

"YES" cut off (message truncated) and discarded,

"NO" kept and indicated (= KEEP), so that the excess portion can be picked up by another YRECEIVE after the CS state has been set (CS="YES" in BEF-NAME).

This item is interpreted for YOPNCON and YCHANGE.

Note

This specification can be changed again in a YRECEIVE (see TRUNC in the instruction structure BEF-NAME).

```
03 SYSCODE PIC X(3). { "YES"
                     "NO" }
```

For NEA transport service only: the message to be received

"YES" is to be converted by the system into EBCDI code if necessary; the system expects messages that are to be sent in EBCDIC,

"NO" is not to be converted but passed in the code of the communication partner. This also applies for the reverse direction (=BINARY: data transfer is code transparent).

This item is interpreted for YOPNCON and YCHANGE, but only if EDIT="USR" has also been specified.

```
03 APPSTART PIC X(3). { "YES" }
                       { "NO" }
```

For NEA transport service only: data transfer is begun by

"YES" this specific DCAM application (PROC=APPSTART: if ACCPT="YES", this is the definitive specification; if ACCPT="NO", it is a suggestion),

"NO" any partner (= ANYSTART: nothing is specified).

This item is interpreted for YOPNCON.

```
02 MAXLN PIC 9(4) COMP.
```

contains the maximum message length which an application wishes to send on this connection. The physical message length must be 9 bytes shorter than specified for MAXLN. DCAM checks the user's specification, reduces it if necessary to the maximum possible message length and writes it back to the MAXLN item.

When you specify MAXLN, be sure to enter the value again after every YOPNCON call with the same connection structure, otherwise the value previously entered by DCAM will be interpreted as the MAXLN specification.

The value entered here is not forwarded to the communication partner. It is used merely for optimizing the buffers provided by the system.

The maximum message length depends on the particular configuration and the system generation.

If the COBRUN parameter "TRUNC=NO" is used, 64535 may be specified as the maximum value.

Further information on the MAXLN function is provided in the "DCAM Program Interfaces" manual.

```
03 EXTND PIC X(3). { "YES"
                    "NO" }
```

For NEA transport service only: the data is output

"YES" protected. Unprotected items must be defined by logical control characters and are all transferred to the computer upon entry. This entry is possible only with EDITOUT="LIN" and for the TD 816x, TD 975x, TD 976x and 3270 terminals All the EDIT options are ignored except BELL, GETFC and LCASE.

"NO" unprotected.

```
03 LOGC PIC X(3). { "YES"
                    "NO" }
```

For NEA transport service only: the logical control characters are

"YES" evaluated,

"NO" not evaluated.

```
03 LACK PIC X(3). { "YES"
                    "NO" }
```

For NEA transport service only: a logical acknowledgment is

"YES" requested,

"NO" not requested.

```
03 SIGNAL PIC X(3). { "YES"
                      "NO" }
```

Data flow control; on clearing a bottleneck

"YES" the user receives a GO signal,

"NO" the user is not notified.

```
03 TERMSTAT PIC X(3). { "YES" }  
                       { "NO" }
```

For NEA transport service only;

"YES" The terminal status is queried (support for the 9763 Data Display Terminal),

"NO" The terminal status is not queried.

This item is interpreted for YOPNCON, YRECEIVE and YSEND. If TERMSTAT="YES", the terminal status is queried for each YOPNCON and once it has arrived a positive acknowledgment is issued with the next YRECEIVE. For each YSEND and YRECEIVE a warning is issued until the terminal status is has arrived.

```
02 MDATA PIC X. { "Y" }  
                 { "N" }
```

For ISO transport service only: the more-data function;

"Y" The more-data function is used for this connection. This defines how the data units are transferred at the user's own local DCAM(ISO) interface, but provides no indication of how the physical data blocks sent to the remote transport system are divided up.

"N" The more-data function is not used for this connection.

This item is interpreted for YOPNCON.

MDATA cannot be modified once a connection has been set up.

Detailed information on the more-data function is provided in the manual "DCAM Program Interfaces".

02 RLTH PIC 9(8) COMP.

For ISO transport service only: specifies the maximum expected message length. Default value: 65535. This value is not forwarded to the communication partner. It is used merely for optimizing the buffers provided by the system. The entry here is dependent on the value in the MDATA item:

- If MDATA="N", the maximum expected message length can be specified in RLTH, thereby saving memory space. However, in this case RLTH offers no guarantee that longer messages will not be received.
- If MDATA="Y", the maximum possible length of the messages to be received will be entered after successful connection setup. Any value specified for RLTH is ignored and overwritten. The communication system ensures that no longer messages arrive.

This item is interpreted for YOPNCON.

RLTH has no influence on the message length in the send direction. (This depends on MAXLN.)

Further information on the RLTH function is provided in the "DCAM Program Interfaces" manual.

02 ROUTN PIC 9(4) COMP. ($0 \leq n \leq 8$)

specifies the number of routes to individual partners.

ROUTN=0 DCAM uses the default route.

ROUTN=1...8 Up to 8 routes can be specified at the COBOL interface. DCAM attempts to set up a connection via the specified routes in the order in which they occur. The application cannot determine which route will be used for the connection if more than one route is specified.

This item is interpreted for YOPNCON.

02 ROUTL PIC X(8). route-name

specifies the names of various routes to the partner (PTNNAME, PRONAME). These parameters are evaluated only if ROUTN>0.

This item is interpreted for YOPNCON.

2.2.5 Distribution structure (VTLG-NAME)



This section applies only to DCAM(NEA) transport service applications.

Function

The position and length of the distribution code in a message, as well as the code groups, are defined in the distribution structure.

Format

Level numbers, record names and data names	PICTURE clause	Parameter input
01 VTLG-NAME		
02 CODEIDEN	PIC X(8).	structure-name
02 CODELN	PIC 9(4) COMP.	(0 < length < 9)
02 CODEPOS	PIC 9(4) COMP.	(0 ≤ position < 256)
02 CODEIND	PIC X.	code-indicator
02 FILLER	PIC X.	
02 CODEGRNO	PIC 9(4) COMP.	(0 < quantity < 9)
02 CODEGROUP	OCCURS 8 TIMES.	
03 CODENAM	PIC X(4).	group-name
03 CODENO	PIC 9(4) COMP.	(0 < amount < 9)
03 CODEVAL	PIC X(8). OCCURS 8 TIMES.	codes

Input

02 CODEIDEN PIC X(8). structure-name

structure-name is the symbolic name of a distribution structure for purposes of identification in subsequent processing. It may consist of any alphanumeric characters and must be entered in the item left-justified; any remaining positions to the right must be filled with blanks (SPACES).

This item is interpreted for YOPNCON, YCHANGE and YPERMIT.

02 CODELN PIC 9(4) COMP. (0 < length < 9)

length specifies the length of the distribution code in the message. A distribution code may be up to 8 characters long. This entry holds for the subsequently defined distribution code(s).

This item is interpreted for YOPNCON.

02 CODEPOS PIC 9(4) COMP. (0 ≤ position < 256)

position specifies the position of the distribution code in the message. The displacement relative to the first character of the message is given. The first character has the displacement 0. The displacement may not

- exceed 255;
- exceed the total length (information and distribution codes) of the message. This entry holds for the subsequently defined distribution code(s).

This item is interpreted for YOPNCON.

02 CODEIND PIC X. code-indicator

code-indicator specifies the character which indicates the distribution code. If the value of this operand does not equal LOW-VALUE, an implicit distribution code can be used. In this case the following applies: if this character is located at CODEPOS, then this is followed by the distribution code, whose length is as specified in CODELN (but no greater than 7 characters). If it is not located, the message is passed to the task that also received the previous message (implicit distribution code).

If LOW-VALUE is specified for CODEIND, a distribution code is explicitly expected in every message.

02 CODEGRNO PIC 9(4) COMP. (0 < quantity < 9)

quantity specifies the number of active distribution code groups. Up to 8 groups may be used.

This item is interpreted for YOPNCON, YCHANGE and YPERMIT.

02 CODENAM PIC X(4). group-name

group-name is the symbolic name of a distribution code group for purposes of identification in subsequent processing. It may consist of any alphanumeric characters and must be entered in the item left-justified; any remaining positions to the right must be filled with blanks (SPACES).

This item is interpreted for YOPNCON, YCHANGE and YPERMIT.

02 CODENO PIC 9(4) COMP. (0 < amount < 9)

amount specifies the number of codes defined in a group. Up to 8 codes may be described in each group.

This item is interpreted for YOPNCON, YCHANGE and YPERMIT.

02 CODEVAL PIC X(8) OCCURS 8 TIMES. code

code specifies the value of the distribution code. It may consist of any alphanumeric characters and must be entered in the item left-justified; any remaining positions to the right must be filled with blanks (SPACES). Codes defined with reference to a structure name (CODEIDEN) must be unique.

This item is interpreted for YOPNCON, YCHANGE and YPERMIT.

2.2.6 Wait structure (WAIT-NAME)

Function

In the wait structure the operands for waiting for termination of asynchronous CALLs are described.

Format

Level numbers, record names and data names	PICTURE clause	Parameter input DCAM(ISO) DCAM(NEA)
01 WAIT-NAME.		
02 LIFETIM	PIC 9(5) COMP.	$\left[\begin{array}{l} (1 \leq n \leq 42300) \\ 60 \end{array} \right]$
02 FILLER	PIC X(12).	
02 DCAM-RETURN-INFO.		
03 RETCODE.		
04 WAIT-RC.		
05 RRS	PIC 9(4) COMP.	
05 RSI	PIC 9(4) COMP.	
03 EREIGNIS	PIC 9.	
88 NOEVENT	VALUE 0.	
88 LETTER	VALUE 1.	
88 OPENED	VALUE 2.	
88 GOSIGNAL	VALUE 3.	
88 LOSCON	VALUE 4.	
03 FILLER	PIC X(3).	

Key:

EREIGNIS = event

Input

```
02 LIFETIM PIC 9(5) COMP. { (1 ≤ n ≤ 42300) }
                        { 60 }
```

n is the time in seconds during which the task is to wait for occurrence of an event. The return information (RETCODE) indicates whether the event has occurred or the wait time has been exceeded.

This item is interpreted for YWAIT.

03 RETCODE.

In this feedback field, the user receives the return code of the wait call. If RSI=0 (event received), the return code indicates the termination of the asynchronous call or the LOSCON reason in the instruction structure specified with the YWAIT request.

03 EREIGNIS PIC 9.

specifies the event identifier:

NOEVENT	no event received,
OPENED	the YOPNCON request has been terminated,
LETTER	the YRECEIVE request has been terminated,
GOSIGNAL	the memory bottleneck has been cleared,
LOSCON	the connection has been cleared down by the system or partner.

2.2.7 Additional transfer areas

2.2.7.1 Standard areas

Additional transfer areas are required for some calls. These areas serve different purposes depending on the use of the call, but they always have the following format:

Format

Level numbers, record names and data names	PICTURE clause	Parameter input
01 BEREICH		
02 LAENGE	PIC 9(4) COMP.	length
02 TEXT	PIC X(..).	data-item

Key:

BEREICH = area
LAENGE = length

Input

02 LAENGE PIC 9(4) COMP. length

length specifies the length of the data item required for the call. The table below shows the valid lengths and calls for which the item is interpreted.

02 DATAITEM PIC ... data-item

data-item specifies the contents of the item. The contents will differ depending on the use, as will the type of item (numeric, alphanumeric) and its subdivision. The table below shows the contents of the individual items.

The calls for which the item is interpreted are also shown in the table.

Length	Contents	Call	Page
2	Application status indication	YINQUIRE	69
for DCAM(ISO): (1 ≤ n ≤ 32) for DCAM(NEA): (1 ≤ n ≤ 80)	Connection message (send data)	YOPNCON	76 and 85
	Connection message (receive data)	YINQUIRE	90
14	Partner data for connection setup		
26	Partner characteristics (NEA transport service only)	YINQUIRE	92
4	No. of connections	YINQUIRE YINQUIRE	100
8	Peripherals (NEA transport service only)	YINQUIRE YINQUIRE	98
24	Basic terminal information (NEA transport service only)	YINQUIRE	96
≤ 14	Terminal and character sets (NEA) transport service only)	YINQUIRE	94
(1 ≤ n ≤ 8)	Express message (NEA transport service only)	YSEND	108
{(1 ≤ n ≤ maxln)}	Message (send data)		
(1 ≤ n ≤ maxln)	Message (receive data)	YRECEIVE	111

Note

Because of their similarity, the data items can be overlaid. For example:

```

01 BEREICH.
  02 LAENGE PIC 9(4) COMP VALUE IS 80.
  02 DATENFELD.
    03 FILLER PIC X(80).
  02 ANW-BEREICH REDEFINES DATENFELD.
    03 ZUSTAND PIC 9(4) COMP.
    03 FILLER PIC X(78).

```

etc.

Key:

```

BEREICH      = area
LAENGE       = length
DATENFELD    = data item
ANW-BEREICH  = application area
ZUSTAND      = state

```

2.2.7.2 Areas for the YOPNCON call

Function

For the YOPNCON call 2 further transfer areas are required. They have the following format:

Format

Level numbers, record names and data names	PICTURE clause	Parameter input
01 BEREICH1.		
02 LAENGE1	PIC 9(4) COMP.	length
02 TEXT1	PIC X(..).	message
01 BEREICH2.		
02 LAENGE2	PIC 9(4) COMP.	length
02 TEXT2	PIC X(..).	data-item

Key:

BEREICH = area
LAENGE = length

BEREICH1

Name of a structure containing the connection message to be sent to the partner.

BEREICH2

Name of the structure to which the connection message from the partner is to be brought.

Note

BEREICH2 is possible only when ACCEPT=NO is specified.

Input

02 LAENGE1 PIC 9(4) COMP. length

length Length of the connection message to be sent to the partner. If no connection message is to be sent to the partner but one is expected from him, the operand must be specified as 0 (serves as dummy parameter).

02 TEXT1 PIC X(..). message

message Message to be sent to the partner.

02 LAENGE2 PIC 9(4) COMP. length

length Length of the area to which the connection message from the partner is to be brought. If the value 0 is specified no message is received. After the call either the actual length of the connection message appears here or the number of positions truncated if the area was too short to receive them.

02 TEXT2 PIC X(..). data-item

data-item Area into which the connection message is to be written by the partner.

2.2.7.3 Single items

Function

Certain calls also require single items.

Format

Level numbers, record names and data names	PICTURE clause	Parameter input	
		DCAM (ISO)	DCAM (NEA)
77 FUNKTION	PIC X(3).	$\left\{ \begin{array}{l} \text{"APP"} \\ \text{"TOP"} \\ \text{"CNT"} \end{array} \right\}$	$\left\{ \begin{array}{l} \text{"APP"} \\ \text{"TOP"} \\ \text{"CNT"} \\ \text{"PTN"} \\ \text{"BTI"} \\ \text{"MCS"} \\ \text{"POT"} \end{array} \right\}$
77 VT-NAME	PIC X(8).	-	distribution-name
77 CG-NAME	PIC X(4).	-	group-name
77 DCAMVER	PIC 9(4) COMP-3.		
88 DCAMV8	VALUE 800.		

Key:

FUNKTION = function

Input

77 FUNKTION PIC X(3). $\left\{ \begin{array}{l} \text{"APP"} \\ \text{"TOP"} \\ \text{"CNT"} \end{array} \right\} / \left\{ \begin{array}{l} \text{"PTN"} \\ \text{"BTI"} \\ \text{"MCS"} \\ \text{"POT"} \end{array} \right\}$

This item specifies one of the different functions of the YINQUIRE call for querying either

- "APP" the status of the DCAM application (APPSTAT),
- "TOP" partner information prior to connection setup (TOPLOGON),
- "CNT" the number of partners (COUNTPTN),
- "PTN" partner characteristics (PTNCHAR) (NEA transport service only),

- "BTI" basic terminal information (BTERMINF) (NEA transport service only),
- "MCS" the display terminal and character sets (MONCHARS) (NEA transport service only),
- "POT" the peripherals (PEROTERM) (NEA transport service only).

Note

The partner characteristics are described on page 135 ff. The status information for the terminal, the peripherals and the basic terminal information are described in the chapter entitled "Status information" in the "VTSU User Guide".

This item is interpreted for YINQUIRE.

77 VT-NAME PIC X(8). distribution-name

distribution-name

is the name of a distribution. The first character must be alphabetic (A-Z, @, #, \$). The remaining seven characters can be alphanumeric (A-Z, 0-9, @, #, \$). The name must be entered in the item left-justified; any remaining positions to the right are filled with blanks (SPACES), if necessary.

This item is interpreted for YPERMIT and YFORBID.

77 CG-NAME PIC X(4). group-name

group-name is the name of a distribution code group. The name must be alphabetic (A-Z) and it must be entered in the item leftjustified; any remaining positions to the right are filled with blanks (SPACES), if necessary.

This item is interpreted for YPERMIT.

77 DCAMVER PIC 9(4) COMP-3.

In this item the functional version of DCAM is specified.

DCAMV8 All functions as of DCAM V8.0 can be used. This entry is necessary if the new functions as of DCAM V8.0 (=0800) are to be used.

2.2.7.4 Areas with control characters



This section applies only to DCAM(NEA) transport service applications.

The message output may be influenced by using numeric constants as control characters if TRANSF="LIN". These control characters are created in their own areas. You will find a list of control characters in the chapter entitled "Logical control characters" in the "VTSU User Guide".

The control characters include

- NEW-LINE
- NEW-PAGE
- CONT-SAME-LINE
- CONT-LINE
- CONT-COL
- SHEET-FEED
- CONT-ACT-POS
- EMPH-LAYOUT1
- EMPH-LAYOUT2
- EMPH-LAYOUT3
- EMPH-LAYOUT4
- NORMAL-LAYOUT
- DARK-LAYOUT
- PART-LINE-UP
- PART-LINE-DOWN
- SECOND-CHAR-SET
- NORMAL-CHAR-SET
- START-PROT-AREA
- END-PROT-AREA
- START-NUM-DATA
- VERT-MOVE-IND
- HORIZ-MOVE-IND
- .
- .
- .

Control characters are inserted in the message at the desired position or the message received can be queried to ascertain the contents of these control items.

2.3 Description of the CALLs (PROCEDURE DIVISION)

The modules implementing the called functions are invoked by the user program in the form of subroutines. COBOL subroutine techniques are described in the relevant COBOL manuals. At this point, certain rules of particular importance to the DCAM user will be briefly recapitulated.

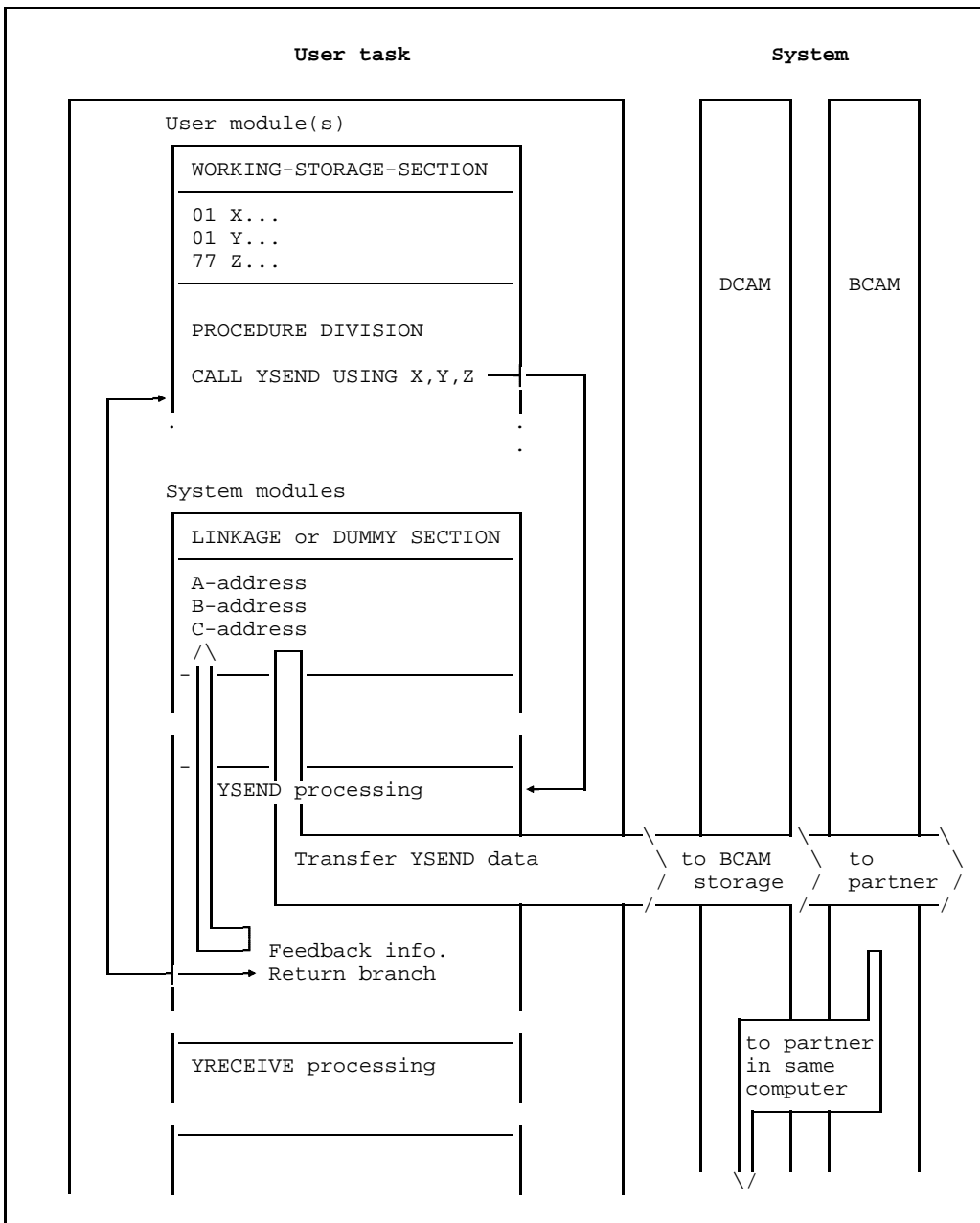
2.3.1 Execution of (subroutine) calls

General

The following diagram presents an example of call execution.

During the linkage editor run, the DCAM module YDDCUROT is linked to the user module(s) from LMS library \$TSOS.SYSLIB.DCAM. When a DCAM module is called, it is loaded dynamically (if this has not already been done) from the same library (fixed name assignment) and a branch is effected within the application program. The DCAM module accesses the data items, finding in the case of YSEND the name of the originator, the address of the recipient, the message text, its length, etc. The module then transfers the message to DCAM using macro calls (see "DCAM Macros"). It enters the feedback information in the data structure and returns control to the user module, which continues with the next instruction after the CALL.

During system module processing, the user module is in the wait state (synchronous processing). It is restarted by the return branch from the DCAM module.



Example of CALL execution

2.3.1.1 Wait times

Considerable delays can arise during execution of YRECEIVE or YOPNCON because it is necessary to wait for the message or connection request, or for the acceptance of the or a partner. Consequently users have two options:

- They can, in issuing the call, **specify a time** after which the call is terminated automatically if nothing has arrived from the partner (see TOVAL and Q in the instruction structure BEF-NAME, page 17).
- They can specify that the call be **terminated at once** even if nothing has arrived from the partner (see Q in instruction structure BEF-NAME, page 17).
In this case the call is issued "tentatively" and may have to be repeated later. Delays in program execution can thus be almost completely avoided, though the overhead for repeated placement of the call becomes a factor to note.

2.3.1.2 Execution with asynchronous DCAM calls

Calls for which it is necessary to wait for an event to occur can also be executed asynchronously. This means that "SYN=NO" must be specified. If the call has been accepted (return code 0) a YWAIT call can be issued.

3 Using the DCAM functions

Before you read this chapter, you must be familiar with the DCAM functions as described in the chapter "DCAM functions" of the "DCAM Program Interfaces" manual. The relevant chapter is organized the same way in both manuals, enabling you to consult "DCAM Program Interfaces" without having to use cross-references.

So you can get to know the interface quickly, particular attention has been paid to presenting it as simply as possible. Schematics have been designed to help you; the figure below shows you how to interpret them.

The diagrams in the individual chapters all follow the same pattern. Mandatory entries are placed before optional ones and the two types of entry appear in the diagram in different colors. Default values are not explicitly stated because they do not have to be entered, provided, of course, the items concerned always have defined initial values, i.e. in this case are blank-or zero-filled and hence reset. This is, however, standard practice and constitutes one of the basic guidelines for good, clear programming. Should you nevertheless want to set default values explicitly, you should refer to the relevant specifications in the section "Description of data structures and transfer areas".

3.1 CALLs relating to the existence of a DCAM application

Existence-oriented calls are used to open or close DCAM applications. They can also be used to query or, in the case of DCAM(NEA) transport service applications, change the status of a DCAM application.

The existence-oriented calls are:

- **YOPEN** **Open a DCAM application**
- **YCLOSE** **Close a DCAM application**
- **YINQUIRE** **Query the status of a DCAM application**



and additionally for DCAM(NEA) transport service applications:

- **YSETLOG** **Change the status of a DCAM application**

3.1.1 Opening a DCAM application

A DCAM application can be

- **non-shareable** (i.e. it can only be opened by one task), or
- **shareable** (i.e. it can be opened by the primary task and by secondary tasks).

If the application is shareable, message distribution can be handled

- using the **standard procedure** (originator-oriented/common receiver), or
- using **distribution codes** (for DCAM(NEA) transport service applications).

These possibilities result in a number of variants of the YOPEN call, which are described below.

3.1.1.1 Opening a non-shareable DCAM application

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;">YOPEN</div>	→01 APP-NAME .			
	02 APPNAME	application-name		1)
	.			
	02 PRONAME	processor-name		2)
	.			
	02 DISNAME			
	02 LINK	link-name		3)
	02 LINKMOD	"PER"		4)
	02 LOGPASS	password1	LOW-VALUE	5)
	02 USEPASS			
	02 USEPW	password3	password2	6)
	02 VERIFY			
02 ATTR.				
03 SHARE				
03 LOGON	"NO"		7)	
03 DISCO				
03 TACK		"NO"	8)	
02 FDBK SYN- CHRONIZED.				
03 RCD		return code	9)	
03 ECD		error code	10)	
03 IND				
02 ISO		"Y"	11)	
.				
→01 DCAMVER		dcam-version	0800	12)

Opening a non-shareable DCAM application

Format

CALL YOPEN USING APP-NAME[DCAMVER].

.
.

.

APP-NAME Name of an application structure.

DCAMVER indicates the functional DCAM version.
If DCAMVER is not specified or is not assigned the value 0800, only the functions of DCAM V7.0 are used.

Mandatory specifications:

- 1) the name of the DCAM application;
- 5) LOW-VALUE in the LOGPASS item (ISO transport service only);
- 8) "NO" in the TACK item (ISO transport service only);
- 11) that this is a DCAM(ISO) transport service application (ISO="Y");
- 12) 0800 as the DCAM version number (ISO transport service only).

Optional specifications:

- 3) the link name if name assignments (LINK) are to be used (see section on "Name assignment function" in the "DCAM Program Interfaces" manual and the /SET-DCAM-APPLICATION-LINK command on page 192);
- 4) that the entries from the CLT (communication link table) are to be available to the program during the entire run (LINKMOD="PER"). If this specification is omitted, they are available only during the call;
- 5) the password to be entered by the partner at the time of the connection request (NEA transport service only). If this specification is omitted, no password is requested;
- 6) the password that was specified for this application at RDF generation;
- 7) that connection requests are not to be processed (LOGON="NO"), (NEA transport service only). The default value is LOGON="YES", i.e. requests are to be processed;
- 12) the DCAM version number, e.g. 0800, if the new functions as of DCAM V8.0 are to be used (NEA transport service only). 0800 must be specified in order to use the new functions. If not specified, or if DCAMVER \neq 0800, a DCAM version \leq 7.0 is assumed.

DCAM then returns:

- 2) the symbolic name of the home processor;
- 9) the return code;
- 10) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.1.1.2 Primary opening of a shareable DCAM application

A shareable DCAM application is to be opened by the primary task.

CALL	USING	Parameter input		Description
		NEA	ISO	
YOPEN	→01 APP-NAME .			
	02 APPNAME .	application-name		1)
	02 PRONAME .	processor-name		2)
	02 DISNAME .			
	02 LINK .	link-name		3)
	02 LINKMOD .	"PER"		4)
	02 LOGPASS .	password1	LOW-VALUE	5)
	02 USEPASS .	password2	password1	6)
	02 USEPW .	password3	password2	7)
	02 VERIFY .		"PRM"	8)
	02 ATTR .		"YES"	9)
	03 SHARE .			
	03 LOGON .	"NO"		10)
	03 DISCO .			
	03 TACK .	{ "REQ" "NO" }	"NO"	11)
02 FDBK SYN- CHRONIZED .				
03 RCD .		return-code	12)	
03 ECD .		error-code	13)	
03 IND .				
02 ISO .		"Y"	14)	
→01 DCAMVER		dcam-version	0800	15)

Primary opening of a shareable DCAM application

Mandatory specifications:

- 1) the name of the DCAM application;
- 5) LOW-VALUE in the LOGPASS item (ISO transport service only);
- 9) that this application is to be shareable;
- 11) "NO" in the TACK item (ISO transport service only);
- 14) that this is a DCAM(ISO) transport service application (ISO="Y");
- 15) 0800 as DCAM version number (ISO transport service only).

Optional specifications:

- 3) the link name if name assignments (LINK) are to be used (see section on "Name assignment function" in the "DCAM Program Interfaces" manual and the /SET-DCAM-APPLICATION-LINK command on page 192);
- 4) that the entries from the CLT (communication link table) are to be available to the program during the entire run (LINKMOD="PER"). If this specification is omitted, they are available only during the call;
- 5) the password to be entered by the partner at the time of the connection request (NEA transport service only). If this specification is omitted, no password is requested;
- 6) the password to be entered when the application is opened by a secondary task. If this specification is omitted, no password is requested. The password can only be specified if there is no RDF password, or if both passwords are identical;
- 7) the password that was specified for this application at RDF generation;
- 8) that the opening task is the primary task (PRIMARY). If this item is blank, no check is carried out (NO);
- 10) that connection requests are not to be processed (LOGON="NO") (NEA transport service only). The default value is LOGON="YES", i.e. requests are to be processed;
- 11) that transport acknowledgments are to be forwarded to the requesting task ("REQ"=REQTASK) or are not to be passed to the application ("NO"=NOTACK) (NEA transport service only). As the default option, they are forwarded to the primary task ("PRI"=PRIMTASK);
- 15) the DCAM version number, e.g. 0800, if the new functions as of DCAM V8.0 are to be used (NEA transport service only). 0800 must be specified in order to use the new functions. If not specified, or if DCAMVER \neq 0800, a DCAM version \leq 7.0 is assumed.

DCAM then returns:

- 2) the symbolic name of the home processor;
- 12) the return code;
- 13) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.1.1.3 Primary opening - use of distribution codes

 This section applies only to DCAM(NEA) transport service applications.

A shareable application is to be opened by the primary task. Message distribution is to be handled using distribution codes.

CALL	USING	Parameter input		Description
		NEA	ISO	
YOPEN	→01 APP-NAME .			
	02 APPNAME	application-name		1)
	.			
	02 PRONAME	processor-name		2)
	.			
	02 DISNAME	distribution-name		3)
	02 LINK	link-name		4)
	02 LINKMOD	"PER"		5)
	02 LOGPASS	password1		6)
	02 USEPASS	password2		7)
	02 USEPW	password3		8)
	02 VERIFY	"PRM"		9)
	02 ATTR.			
	03 SHARE	"YES"		10)
	03 LOGON	"NO"		11)
	03 DISCO	"YES"		12)
03 TACK	{ "REQ" "NO" }		13)	
02 FDBK SYN- CHRONIZED.				
03 RCD	return-code		14)	
03 ECD	error-code		15)	
03 IND				
.				
→01 DCAMVER		dcam-version		16)

Primary opening - use of distribution codes

Mandatory specifications:

- 1) the name of the DCAM application;
- 3) the distribution name under which this task can be assigned a distribution code group;
- 10) that this application is to be shareable;
- 12) that messages are to be distributed via distribution codes.

Optional specifications:

- 4) the link name if name assignments (LINK) are to be used (see section on "Name assignment function" in the "DCAM Program Interfaces" manual and the /SET-DCAM-APPLICATION-LINK command on page 192);
- 5) that the entries from the CLT (communication link table) are to be available to the program during the entire run (LINKMOD="PER"). If this specification is omitted, they are available only during the call;
- 6) the password to be entered by the partner at the time of the connection request. If this specification is omitted, no password requested;
- 7) the password to be entered when the application is opened by a secondary task. If this specification is omitted, no password is requested. The password can only be specified if there is no RDF password, or if both passwords are identical;
- 8) the password that was specified for this application at RDF generation;
- 9) that the opening task is the primary task (PRIMARY). If this entry is blank, no check is carried out (NO);
- 11) that connection requests are not to be processed (LOGON="NO"). The default value is LOGON="YES", i.e. requests are to be processed;
- 13) that transport acknowledgments are to be forwarded to the requesting task ("REQ"=REQTASK) or are not to be passed to the application ("NO"=NOTACK). As the default option, they are forwarded to the primary task ("PRI"=PRIMTASK);
- 16) the DCAM version number, e.g. 0800, if the new functions as of DCAM V8.0 are to be used. 0800 must be specified in order to use the new functions (for NEA transport service only). If not specified, or if DCAMVER \neq 0800, a DCAM version \leq 7.0 is assumed.

DCAM then returns:

- 2) the symbolic name of the home processor;
- 14) the return code;
- 15) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177).

3.1.1.4 Secondary opening

A DCAM application that has already been opened by a primary task is to be opened by a secondary task (i.e. subsequently).

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;">YOPEN</div>	→01 APP-NAME .			
	02 APPNAME .	application-name		1)
	02 PRONAME .	processor-name		2)
	02 DISNAME .			
	02 LINK .	link-name		3)
	02 LINKMOD .	"PER"		4)
	02 LOGPASS .			
	02 USEPASS .			
	02 USEPW .	password3		5)
	02 VERIFY .	"SEC"		6)
	02 ATTR .			
03 SHARE .	"YES"		7)	
03 LOGON .				
03 DISCO .				
03 TACK .				
02 FDBK SYN-CHRONIZED .				
03 RCD .	return code		8)	
03 ECD .	error-code		9)	
03 IND .				
02 ISO .		"Y"	10)	
→01 DCAMVER	dcam-version	0800	11)	

Secondary opening

Mandatory specifications:

- 1) the name of the DCAM application;
- 7) that this application is to be shareable;
- 10) that this is a DCAM(ISO) transport service application (ISO="Y");
- 11) 0800 as the DCAM version number (ISO transport service only).

Optional specifications:

- 3) the link name if name assignments (LINK) are to be used (see section on "Name assignment function" in the "DCAM Program Interfaces" manual and the /SET-DCAM-APPLICATION-LINK command on page 192);
- 4) that the entries from the CLT (communication link table) are to be available to the program during the entire run (LINKMOD="PER"). If this specification is omitted, they are available only during the call;
- 5) the password specified by the primary task for connection to the application or the password that was specified for this application at RDF generation;
- 6) that the opening task is the secondary task (SEC). If this item is blank, no check is carried out (NO);
- 11) the DCAM version number, e.g. 0800, if the new functions as of DCAM V8.0 are to be used (NEA transport service only). 0800 must be specified in order to use the new functions. If not specified, or if DCAMVER \neq 0800, a DCAM version \leq 7.0 is assumed.

Note

In the case of a secondary YOPEN for an application, the same DCAMVER must be specified as for the primary YOPEN.

DCAM then returns:

- 2) the symbolic name of the home processor;
- 8) the return code;
- 9) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.1.1.5 Secondary opening - use of distribution codes



This section applies only to DCAM(NEA) transport service applications.

A DCAM application that has already been opened by a primary task is to be opened by a secondary task (i.e. subsequently). As defined in the primary task, message distribution is to be handled using distribution codes.

CALL	USING	Parameter input		Description
		NEA	ISO	
YOPEN	→01 APP-NAME .			
	02 APPNAME	application-name		1)
	02 PRONAME	processor-name		2)
	02 DISNAME	distribution-name		3)
	02 LINK	link-name		4)
	02 LINKMOD	"PER"		5)
	02 LOGPASS			
	02 USEPASS			
	02 USEPW	password3		6)
	02 VERIFY	"SEC"		7)
	02 ATTR.			
03 SHARE	"YES"		8)	
03 LOGON				
03 DISCO				
03 TACK				
02 FDBK SYN- CHRONIZED.				
03 RCD	return-code		9)	
03 ECD	error-code		10)	
03 IND				
← - - -	→01 DCAMVER	dcam-version		11)

Secondary opening - use of distribution codes

Mandatory specifications:

- 1) the name of the DCAM application;
- 3) the distribution name under which this task can be assigned a distribution code group;
- 8) that this application is to be shareable.

Optional specifications:

- 4) the link name if name assignments (LINK) are to be used (see section on "Name assignment function" in the "DCAM Program Interfaces" manual and the /SET-DCAM-APPLICATION-LINK command on page 192);
- 5) that the entries from the CLT (communication link table) are to be available to the program during the entire run (LINKMOD="PER"). If this specification is omitted, they are available only during the call;
- 6) the password specified by the primary task for connection to this application or the password that was specified for this application at RDF generation;
- 7) that the opening task is the secondary task (SEC). If this item is blank, no check is carried out (NO);
- 11) the DCAM version number, e.g. 0800, if the new functions as of DCAM V8.0 are to be used. 0800 must be specified in order to use the new functions (for NEA transport service only). If not specified, or if DCAMVER \neq 0800, a DCAM version \leq 7.0 is assumed.

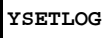
DCAM then returns:

- 2) the symbolic name of the home processor;
- 9) the return code;
- 10) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.1.2 Changing the state of a DCAM application

 This section applies only to DCAM(NEA) transport service applications.

For this function, the DCAM application must be able to process connection requests, i.e. LOGON="YES" must have been specified when it was opened (cf. page 57). After that, it is in the START state.

CALL	USING	Parameter input		Description
		NEA	ISO	
	→01 APP-NAME. 02 APPNAME . .	application-name		1)
	→01 BEF-NAME. . 03 STP . 03 FDBK 04 RCD 04 ECD 04 IND	"NO" return-code error-code		2) 3) 4)

Changing the state of a DCAM application

Mandatory specification:

- 1) the name of the DCAM application for APP-NAME.

Optional specification:

- 2) that the START state should be restored (STP="NO") if the application had previously been placed in the STOP state. If this item is blank, the application is placed in the STOP state (STP="YES" : default value).

DCAM then returns:

- 3) the return code;
- 4) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.1.3 Querying the state of a DCAM application

You can query the state of a DCAM application that you yourself have opened or any other application opened in the same host processor, provided that you have opened at least one application in that processor.

CALL	USING	Parameter input		Description
		NEA	ISO	
YINQUIRE	→77 FUNKTION		"APP"	1)
	→01 APP-NAME. 02 APPNAME .		application-name	2)
	02 FDBK SYN- CHRONIZED. 03 RCD		return-code	3)
	03 ECD		error-code	4)
	03 IND . .			
	→01 CONN-NAME.		CONN-NAME must be specified because this call is also used for the connection function. It is entered here for compatibility reasons only; its contents are not interpreted.	
→01 ANW-BEREICH. 02 LAENGE 02 ZUSTAND		2	state indicator	5) 6)

Querying the state of a DCAM application

Key:

- FUNKTION = function
- ANW-BEREICH = application area
- LAENGE = length
- ZUSTAND = state

Mandatory specifications:

- 1) that it is the state of a DCAM application that is to be queried, since the YINQUIRE call is also used for other queries (see page 89);
- 2) the name of the DCAM application whose state is to be queried;
- 5) the size of the area into which DCAM is to enter the information.

DCAM then returns:

- 3) the return code;
- 4) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff);
- 6) a condition code indicating the state of the DCAM application (for the meaning of the codes, see the following table).

Condition code	ISO	NEA	Meaning
0	x	x	The application has been opened and is processing connection requests (LOGON and START).
2		x	The application has been opened but is not at the moment processing connection requests (LOGON and STOP).
4		x	The application has been opened but does not process connection requests (NLOGON).
6	x	x	The application has not yet been generated (opened by a primary task), i.e. it does not exist.

3.1.4 Closing a DCAM application

Closing a DCAM application by means of a **primary task**:

- The DCAM application is canceled, i.e. it is closed to this task as well as to all secondary tasks belonging to this application.
- All connections existing at this moment are cleared down.
- Pending DCAM function requests are saved.
- Data which has been received but not yet transferred to the task is lost.
- Connection requests queued up at this time will be rejected by DCAM.

This YCLOSE affects the **secondary tasks concerned** as follows:

- The next call using the name of this application will be rejected with the appropriate error code (see page 177ff).

Closing a DCAM application by means of a **secondary task**:

- The DCAM application is closed to this task alone.

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; padding: 2px; display: inline-block;"> YCLOSE </div>	→01 APP-NAME.	application-name		1)
	02 APPNAME			
	02 FDBK SYN- CHRONIZED.			
	03 RCD	return-code		2)
	03 ECD	error-code		3)
	03 IND			

Closing a DCAM application

Mandatory specification:

- 1) the name of the DCAM application to be closed.

DCAM then returns:

- 2) the return code;
- 3) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.2 Connection-oriented CALLs

Before communication partners are able to transfer messages, they must first set up a connection. One of the partners places a request to set up the connection. This request is then transmitted to the other partner by the communication system.

The latter can ask for information about the requesting partner and then accept the connection request.

Certain connection characteristics can be changed after the connection has been established. A connection can be cleared down by any partner at any time.

The connection-oriented calls are:

- **YOPNCON** **Open a connection (request/accept),** and (for DCAM(NEA) transport service applications): Link to a predefined connection.
- **YINQUIRE** **Query the entries on the connection and the connection partner.**
- **YREJLOG** **Reject a connection request.**
- **YCHANGE** **Change the characteristics of a connection.**
- **YCLSCON** **Cancel a request or close a connection.**

In shareable applications these calls, with the exception of YINQUIRE, can only be issued by the primary task.

3.2.1 Establishing a connection

A connection can be established by

- issuing a connection request,
- accepting a connection request.

When establishing a connection, the type of queue (originator-oriented or common receiver) or, for DCAM(NEA) transport service applications, the assignment of distribution codes to the connection is defined.

This results in a number of variants of the YOPNCON call. The connection to be set up is described in the connection structure.

3.2.1.1 Definition of the connection to be established

The connection structure with all its items has already been dealt with in the section "Connection structure" on page 27ff. The items of importance when establishing a connection will now be described in greater detail; items which are handled differently for individual calls are dealt with under the relevant call.

CALL	USING	Parameter input		Description
		NEA	ISO	
For calls, see relevant sections	01 CONN-NAME.			
	02 PTNNAME	} see descriptions of individual calls		1)
	02 PRONAME			2)
	02 PASSWORD			3)
	02 LINK		link-name	4)
	02 LINKMOD		"PER"	5)
	02 EDITIN.			
	03 TRANSF	{ "PHY" "FOR" }		6)
	03 GETBS	"YES"		7)
	03 GETFC	"YES"		7)
	03 LCASE	"YES"		7)
	02 EDITOUT.			
	03 TRANSF	{ "PHY" "FOR" }		8)
	03 HCOPI	"YES"		9)
	03 HOM	"YES"		9)
	02 EDIT	"SYS"		10)
	02 PROC.			
	03 TRUNC		"NO"	11)
	03 SYSCODE	"NO"		12)
	03 APPSTART	"YES"		13)
02 MAXLN		maxln	14)	
02 PTNCHAR.	} see descriptions of individual calls			
03 PTNTYPE				
03 DEFTYPE				
03 CHARSET				
.				
.				
02 PROC1				
03 SIGNAL		"YES"	15)	
03 TERMSTAT	{ "YES" "NO" }		16)	
02 MDATA		"Y"	17)	
02 RLTHA		length	18)	
02 ROUTN		0 ≤ n ≤ 8	19)	
02 ROUTL		route-name	20)	

Definition of the connection to be established

When describing a connection the following specifications are **optional**:

- 4) the link name if name assignments (LINK) are to be used (see section on "Name assignment function" in the "DCAM Program Interfaces" manual and the /SET-DCAM-APPLICATION-LINK command on page 192);
- 5) that the entries from the CLT (communication link table) are to be available to the program during the entire run (LINKMOD="PER"). If this specification is omitted, they are available only during the call;
- 6) the type of virtual terminal for receiving the messages (NEA transport service only):

"PHY" (= PHYS: not a virtual terminal) means that the message is transmitted as it was sent from the terminal. It still contains all the control characters required for the control of the device functions. A message header in the device code (usually ISO 7-bit code) is added to the message (see "9749, 9750, 9752 Data Display Terminals, Programmer's Interface"). The length of the header is indicated in the start-of-message character or is preset for the terminal type.

"FOR" (= FORM: format terminal) means that format handling is used.

If this specification is omitted, "LIN" (= LINE: line terminal) is assumed;

- 7) input message handling (NEA transport service only):
 - backspace characters are not to be interpreted and transmitted (for line or format terminals, they are only interpreted with the 8103 Printer Terminal).
 - the first message character to be transmitted is the function key code.
 - lowercase characters are to be transmitted (for line and format terminals).

If this specification is omitted, backspace characters are interpreted and not transmitted ("NO"=NGETBS), the function key code is not be transmitted ("NO"=NGETFC) and lowercase characters are converted to upper case ("NO"=NLCASE);

- 8) the type of virtual terminal for transmitting the messages (NEA transport service only):

"PHY" (=PHYS: not a virtual terminal) means that the appropriate control characters for the device are added to the message. If a message header is required (see '9749, 9750, 9752 Data Display Terminals, Programmer's Interface"), it must precede the message in the device code. The first character must be the header length byte; this byte is mandatory, but may be specified as length 1.

"FOR" (= FORM: format terminal) means that format handling is used.

If this entry is omitted, "LIN" (= LINE: line terminal) will be assumed;

- 9) handling of messages that have been passed to the system for transmission (NEA transport service only):
 - Where TRANSF="LIN" or "PHY", and for 8151, 8152, 816x and 9750 Data Display Terminals, messages are also to be printed out on an attached hardcopy device (HCOPY). If this entry is omitted, messages are not printed ("NO"=NHCOPY). For this function such a printer must be available. If necessary, you should check this beforehand using YINQUIRE (page 92).
 - With TRANSF="LIN", the message is to be output unstructured, i.e. the entire message is a structural unit. With 816x Data Display Terminals, for example, this corresponds to the setting 'Item' = message. If this entry is omitted, each logical line of a message is regarded as a structural unit ("NO"=NHOM);
- 10) that message editing is to be executed as specified in the EDITINor EDITOUT items (NEA transport service only).

If virtual terminals are to be used, this item is mandatory; otherwise it is only advisable if the communication partner is a terminal.

This item must be blank if the partner is a DCAM application. In this case, "USR"=USER is specified;
- 11) that if a message is too long, the excess portion is held for a further YRECEIVE ("NO"=KEEP). For this function, the CS state must have been set for the connection. If this entry is omitted, the rest of the message is discarded ("YES"=TRUNC). In the connection structure this setting is only valid if it does not conflict with the TRUNC entry in the instruction structure for YRECEIVE (see page 111);
- 12) that messages are to be transmitted code-transparent and passed on as bit patterns (BINARY) without conversion, if the default value ("USR"=USER) has been set for 10) (NEA transport service only).

If this entry is omitted, messages are converted ("YES"=SYSCODE), i.e. received or passed on in EBCDIC;
- 13) that data transmission be started by this application (APPSTART) (NEA transport service only). Without this entry, nothing is specified ("NO"=ANYSTART);
- 14) the maximum message length which an application wishes to send on this connection. This **must** be specified if functions as of DCAM V8 are to be used;

- 15) that when a bottleneck is cleared (message flow control) the application receives a GO signal (SIGNAL="YES"). If this is not specified, the application is not notified;
- 16) whether the terminal state is to be queried (NEA transport service only). If TERMSTAT="YES", the terminal state is queried at each YOPNCON and when received is positively acknowledged with the next YRECEIVE. Until it is received, each YSEND/YRECEIVE is accompanied by a warning that a message edited by VTSU exists without a status response;
- 17) that the more-data function is to be used for this application (MDATA="Y") (ISO transport service only);
- 18) the maximum expected message length if MDATA="N"; or the maximum possible message length if MDATA="Y" (entered by DCAM after successful connection setup) (ISO transport service only);
- 19) the number of routes to specific partners;
- 20) the route names.

3.2.1.2 Connection request

The YOPNCON call is used to request that a connection be established.

For DCAM(NEA) transport service applications, this variant applies when distribution codes are not being used. It is assumed that when the DCAM(NEA) transport service application was opened, the attribute NSHARE (page 58) or SHARE with NDISCO (SHARE="YES" and DISCO="NO", page 60) have been entered in the APP-NAME.

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; display: inline-block; padding: 2px;">YOPNCON</div>	→01 APP-NAME. 02 APPNAME .	application-name		1)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME 02 PASSWORD .	partner-name processor-name password4 for further entries see page 73		2) 2) 3)
	02 EDIT .	{"USR" "SYS"}		4)
	03 APPSTART 02 MAXLN 02 PTNCHAR.	{"YES" "NO"} maxln	maxln	5) 6)
	03 PTNTYPE 03 DEVTYPE 03 CHARSET 03 DEVSEC .	First four items of the partner characteristics		} 7)
	02 MDATA 02 RLTH		"Y" length	8) 9)
	→01 BEF-NAME. .			
	02 OPTCD 03 SPEC 03 CS .		"YES" "YES"	10) 11)
	03 ACCPT .		"NO"	12)
	03 FDBK 04 RCD 04 ECD 04 IND .		return- code error-code	13) 14)
	03 SYN		"NO"	15)
	- - - - - →01 VTLG-NAME. .			
	- - - - - →01 BEREICH1. 02 LAENGE1 02 TEXT1		length message	16) 17)
	- - - - - →01 BEREICH2. 02 LAENGE2 02 TEXT2	length data-item	length data-item	18) 19)

Connection request

Key:

BEREICH = area
LAENGE = length

Mandatory specifications:

- 1) the name - in the application structure (APP-NAME) - of the DCAM application which issued the request;
- 2) the address - in the connection structure - of the communication partner. This address consists of the partner name and the name of the processor to which the partner is connected;
- 6) the maximum message length which an application wishes to send on this connection when the new functions as of DCAM V8.0 are to be used (DCAMVER='0800') (NEA transport service only);
- 10) that the request be addressed to a specific partner;
- 12) that a request is to be issued.

Optional specifications:

- 3) a password for setting up a connection, if the partner so desires (NEA transport service only);
- 6) the maximum message length which an application wishes to send on this connection (ISO transport service only);
- 8) that the more-data function is to be used for this connection (MDATA="Y") (ISO transport service only);
- 9) the maximum expected message length for MDATA="N" (ISO transport service only);
- 11) that the originator-oriented queue is to be used. If this entry is omitted, the common receiver queue is automatically used (CA state);
- 15) whether the call is to be executed asynchronously (SYN="NO"). For further details, see the description of the YWAIT call (page 123ff).

Additionally, a connection message may also be transmitted. In this case, the name of a distribution structure must be specified.

The following must then be specified in addition:

- 16) the length of the connection message, which may be between 1 and 32 characters (ISO) or between 1 and 80 characters (NEA);
- 17) the text of the connection message, which must be contained in an item at least as long as the length given in 16);

- 18) the length of the data item to which the connection message is to be sent (NEA transport service only). It may include between 1 and 80 characters;
- 19) the text of the connection message (NEA transport service only), which must be contained in an item at least as long as the length given in 18).

DCAM then returns what the partner defined when accepting the request:

- 4) the type of message editing (NEA transport service only);
- 5) the start of data transmission (NEA transport service only);
- 6) the maximum length of messages to be transmitted on this connection (maximum packet length, see the manual "Generating a Data Communication System"), or system constant if the partner is attached to the same host computer;
- 7) partner characteristics (see page 135) (NEA transport service only);
- 9) the maximum possible data length for MDATA="Y" (ISO transport service only);
- 18) the length of the data item to which the received message is to be transmitted (ISO transport service only). It may include between 1 and 32 characters;
- 19) the text of the received message (ISO transport service only). It must be contained in an item at least as long as the length specified in 18),

and also:

- 13) the return code;
- 14) the error code if the call could not be executed or an error occurred during execution. In this case, 4) through 6) are not returned.

3.2.1.3 Accepting a request

The items in the data structures that are supplied with parameters differ according to whether a request is to be accepted from a specific partner or from any partner.

For DCAM(NEA) transport service applications the following applies:

A connection request need only be accepted if the relevant application is processing connection requests (LOGON specified on opening the application and START state). The variant of YOPNCON to be described this section presupposes that the application has been opened either using the NSHARE attribute (page 58) or the SHARE and NDISCO attributes (SHARE="YES" and DISCO="NO", page 60).

Accepting a request from any partner

CALL	USING	Parameter input		Description
		NEA	ISO	
YOPNCON	→01 APP-NAME. 02 APPNAME .		application-name	1)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME .		partner-name processor-name	2) 2)
	02 MAXLN 02 PTNCHAR. 03 PTNTYPE 03 DEVTYPE 03 CHARSET 03 DEVSEC .		maxln	3)
	02 MDATA 02 RLTH		"Y" length	5) 6)
	→01 BEF-NAME. 02 TOVAL 02 SEQNO 02 OPTCD 03 SPEC 03 CS 03 Q .		(0<n≤32767)	7)
	03 FDBK. 04 RCD 04 ECD 04 IND .		return-code error-code	10) 11)
	03 SYN		"NO"	12)
	→01 VTLG-NAME. .			
	→01 BEREICH1. 02 LAENGE1 02 TEXT1		length message	13) 14)
			For user entries see page 73	
			First four items of the partner characteristics	4)

Accepting a request from any partner

Key:

BEREICH = area
LAENGE = length

Mandatory specification:

- 1) the name of the DCAM application that will accept the request.

Optional specifications:

- 5) that the more-data function is to be used for this connection (MDATA="Y") (ISO transport service only);
- 6) the maximum expected message length for MDATA="N" (ISO transport service only);
- 7) the duration of this call, if it is to be queued. The duration must be at least one second and at most 32767 seconds;
- 8) that the originator-oriented queue is to be used. If this entry is omitted, the common receiver queue is used (CA state);
- 9) that the call is to be queued if it cannot be carried out immediately because no request is pending. If this entry is omitted, the call is terminated at once and must be repeated later if necessary;
- 12) whether the call is to be executed asynchronously (SYN="NO"). For further details, see the description of the YWAIT call, page 123ff.

Additionally, a connection message may also be transmitted. In this case, the name of a distribution structure must be specified.

The following must then be specified in addition:

- 13) the length of the connection message, which may be between 1 and 32 characters (ISO) or between 1 and 80 characters (NEA);
- 14) the text of the connection message, which must be contained in an item at least as long as the length given in 13).

DCAM then returns:

- 2) the address of the communication partner, consisting of the partner and processor name;
- 3) the maximum length of messages to be transmitted on this connection;
- 4) partner characteristics (see page 135) (NEA transport service only);
- 6) the maximum possible message length if MDATA="Y" (ISO transport service only),

and also:

- 10) the return code;
- 11) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

Accepting a request from a specific partner

CALL	USING	Parameter input		Description
		NEA	ISO	
YOPNCON	→01 APP-NAME. 02 APPNAME .		application-name	1)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME .		partner-name processor-name	2) 2)
	02 MAXLN 02 PTNCHAR. 03 PTNTYPE 03 DEVTYPE 03 CHARSET 03 DEVSEC .		maxln	3)
	02 MDATA 02 RLTH		"Y" length	5) 6)
	→01 BEF-NAME. 02 TOVAL 02 SEQNO 02 OPTCD 03 SPEC 03 CS 03 Q .		(0<n≤32767)	7)
	03 FDBK. 04 RCD 04 ECD 04 IND .		return-code error-code	11) 12)
	03 SYN		"NO"	13)
	→01 VTLG-NAME. .			
	→01 BEREICH1. 02 LAENGE1 02 TEXT1		length message	14) 15)

For further entries see page 73

First four items of the partner characteristics

Accepting a request from a specific partner

Key:

BEREICH = area
LAENGE = length

Mandatory specifications:

- 1) the name of the DCAM application that will accept the request;
- 2) the address of the communication partner, consisting of the partner name and the processor name;
- 8) that the request of a specific partner is to be accepted.

Optional specifications:

- 5) that the more-data function is to be used for this connection (MDATA="Y") (ISO transport service only);
- 6) the maximum expected message length for MDATA="N" (ISO transport service only);
- 7) the wait time for this call, if it is to be queued. The wait time must be at least 1 second and at most 32767 seconds. The default value is 600 seconds;
- 9) that the originator-oriented queue is to be used (CS state of the connection). If this entry is omitted, the common receiver queue (CA state) is to be used. This entry is not necessary when using distribution codes;
- 10) that the call is to be queued if it cannot be executed immediately because no request is pending. If this entry is omitted, the call is terminated at once and must be repeated later if necessary;
- 13) whether the call is to be executed asynchronously (SYN="NO"). For further details see the YWAIT call, page 123ff.

Additionally, a connection message may also be transmitted. In this case, the name of a distribution structure must be specified.

The following must then be specified in addition:

- 14) the length of the connection message, which may be between 1 and 32 characters (ISO) or between 1 and 80 characters (NEA);
- 15) the text of the connection message, which must be contained in an item at least as long as the length given in 14).

DCAM then returns:

- 3) the maximum length of messages to be transmitted on this connection;
- 4) partner characteristics (see page 135) (NEA transport service only);
- 6) the maximum possible message length if MDATA="Y" (ISO transport service only),

and also:

- 11) the return code;
- 12) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff). In this case, 3) and 4) are not returned.

3.2.1.4 Connection setup - use of distribution codes

 This section applies only to DCAM(NEA) transport service applications.

The DCAM application must have been opened with the attributes SHARE and DISCO (SHARE="YES" and DISCO="YES", see page 62). Only those parts of a request or acceptance call that are relevant to the use of distribution codes are shown below. It should be noted that the CS entry is not interpreted and can therefore be omitted.

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; padding: 2px; display: inline-block;">YOPNCON</div>	→01 APP-NAME. .	} see page 73		
	→01 CONN-NAME. .			
	→01 BEF-NAME. .	} see page 80		
	→01 VTLG-NAME. 02 CODEIDEN 02 CODELN 02 CODEPOS 02 CODEIND 02 CODEGRNO 02 CODEGROUP 03 CODENAM 03 CODENO 03 CODEVAL			
	-----→01 V-MESSAGE. .	see page 76		

Connection setup - use of distribution codes

The following entries must be made in the **distribution structure**:

- 1) the symbolic name of the distribution structure;
- 2) the length of the distribution codes used on this connection - at least 1 and at most 8 characters;
- 3) the position of the distribution code in the message (0 means as of the first character; 255 is the maximum value);
- 4) whether an implicit distribution code is to be used.
 - If so, a code-indicator character must be defined.
 - If not, the item must be filled with LOW-VALUE;
- 5) the number of attached distribution code groups - at least 1 and at most 8 groups;
- 6) the symbolic name of the distribution code group(s);
- 7) the number of codes contained in a group - at least 1 and at most 8;
- 8) the codes themselves as they are expected by the communication partner.

Note

Only the values of distribution codes can be changed during the life of a connection. This is done using the YCHANGE call (page 103). The entries 7) and 8) are only interpreted if this structure had not already been used for another YOPNCON or YPERMIT (see page 120).

3.2.1.5 Linkage to a predefined connection



This section applies only to DCAM(NEA) transport service applications.

This call is a variant of the calls described on page 76 (without distribution code) and page 85 (with distribution code).

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;">YOPNCON</div>	→01 APP-NAME .			
	02 APPNAME	application-name		1)
	.			
	→01 CONN-NAME .			
	02 PTNNAME	partner-name		2)
	02 PRONAME	processor-name		2)
	.			
	02 EDIT	{"USR" "SYS" }		3)
	.			
	03 APPSTART	{"YES" "NO" }		4)
	02 MAXLN	maxln		5)
	02 PTNCHAR .			
	03 PTNTYPE	} first four items of the partner characteristics		} 6)
	03 DEVTYPE			
	03 CHARSET			
	03 DEVSEC			
	.			
	→01 BEF-NAME .			
	.			
	02 OPTCD .			
03 SPEC	"YES"		7)	
03 CS	"YES"		8)	
.				
03 ACCPT	"NO"		9)	
.				
03 FDBK .				
04 RCD	return-code		10)	
04 ECD	error-code		11)	
04 IND				

Linkage to a predefined connection

Mandatory specifications:

- 1) the name of the DCAM application in the application structure (APP-NAME) wishing to link;
- 2) the address of the communication partner in the connection structure, consisting of the partner name and name of the processor to which the partner is connected;
- 7) that a particular partner is intended;
- 9) that the link is to be with a predefined connection.

Optional specification:

- 8) that the originator-oriented queue is to be used. If this entry is omitted, the common receiver queue (CA state) is used.

DCAM then returns, as specified when the connection was generated:

- 3) the type of message editing;
- 4) the start of data transmission;
- 5) the maximum length of messages to be transmitted on this connection (maximum packet length, see the manual "Generating a Data Communication System"), or system constant if the partner is attached to the same host;
- 6) partner characteristics, see page 135,

and also:

- 10) the return code;
- 11) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff). In this case, 3) through 5) are not returned.

When processing by means of distribution codes, the information given on page 85 also applies.

3.2.2 Querying partner and connection entries

The entries concerning a partner or connection may be queried for various reasons both when establishing a connection and later:

- The **first entry** in an existing request queue is to be queried before connection setup (TOPLOGON).
- The **number** of partners connected or requesting connection is required (COUNTPTN).

and additionally for DCAM(NEA) transport service applications:

The partner characteristics (PTNCHAR) and information about the terminal (BTERMINF), data display terminal, character sets (MONCHARS) and peripherals (PEROTERM) are to be interpreted (see page 135ff). The partner characteristics are described on page 135 ff. The status information for the terminal, the peripherals and the basic terminal information are described in the chapter entitled "Status information" in the "VTSU User Guide".

The YINQUIRE call is used for these various queries.

3.2.2.1 Partner information prior to connection setup (TOPLOGON)

CALL	USING	Parameter input		Description
		NEA	ISO	
YINQUIRE	→77 FUNKTION		"TOP"	1)
	→01 APP-NAME. 02 APPNAME		application-name	2)
	02 FDBK SYN- CHRONIZED.			
	03 RCD		return-code	3)
	03 ECD		error-code	4)
	03 IND			
	→01 CONN-NAME. 02 PTNNAME		partner-name	5)
	02 PRONAME		processor-name	5)
	.			
	→01 TOPL-BEREICH. 02 TOPL-LAENGE		14	6)
	02 PTNCHAR. 03 PTNTYPE		first four items of the partner characteristics	} 7)
	03 DEVTYPE			
03 CHARSET				
03 DEVSEC				
02 PEDIT		{"USR" "SYS"}	8)	
02 PPROC		{"YES" "NO"}	9)	
→01 LGMSG-BEREICH 02 LGMSG-LAENGE		length	10)	
		actual-length	11)	
02 LGMSG		connection-message	12)	

Partner information prior to connection setup

Key:

BEREICH = area
LAENGE = length

Mandatory specifications:

- 1) the function of the call;
- 2) the name of the DCAM application;
- 6) the length of the area in which DCAM is to enter the return information;
- 10) the length of the expected connection message (maximum of 32 bytes for DCAM(ISO) transport service applications or 80 bytes for DCAM(NEA) transport service applications).

DCAM then returns:

- 3) the return code;
- 4) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff). In this case there is no further return information;
- 5) the address of the partner requesting connection, consisting of the partner name and processor name;
- 7) partner characteristics (see page 135) (NEA transport service only);
- 8) the type of message editing proposed by the partner (NEA transport service only);
- 9) the start of data transmission as proposed by the partner (NEA transport service only).

Additionally, if the partner has transmitted this information:

- 11) the actual length of the transmitted connection message;
- 12) the connection message itself.

3.2.2.2 Partner characteristics (PTNCHARS)



This section applies only to DCAM(NEA) transport service applications.

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; padding: 2px; display: inline-block;">YINQUIRE</div>	→77 FUNKTION	"PTN"		1)
	→01 APP-NAME. 02 APPNAME	application-name		2)
	02 FDBK SYN- CHRONIZED.			
	03 RCD	return-code		3)
	03 ECD	error-code		4)
	03 IND			
	→01 CONN-NAME. 02 PTNNAME	partner-name		5)
	02 PRONAME	processor-name		5)
	.			
	→01 PTN-BEREICH. 02 PTN-LAENGE	26		6)
	02 PTNCHAR1.			
	03 PTNTYPE	} first four items of the partner characteristics		} 7)
	03 DEVTYPE			
	03 CHARSET			
	03 DEVSEC			
	02 PTNCHAR2.			
	03 PHYS-EIGEN- SCHAFT.	} physical device characteristics		} 7)
	04 ZEILEN- LAENGE-P			
	04 ZEILEN- ANZAHL-P			
	04 MAXIMAL- PUFFER-P			
	03 LINE-EIGEN- SCHAFT.			
	04 ZEILEN- LAENGE-L	} characteristics of the line terminal		} 7)
	04 ZEILEN- ANZAHL-L			
	04 MAXIMAL- PUFFER-L			
	03 EDITIN	} message editing capability and specifications		}
03 EDITOUT				

Partner characteristics

Key:

FUNKTION	=	function
BEREICH	=	area
LAENGE	=	length
EIGENSCHAFT	=	characteristics
ZEILE	=	line
ANZAHL	=	quantity
MAXIMAL	=	maximum
PUFFER	=	buffer

Mandatory specifications:

- 1) the function of the call;
- 2) the name of the DCAM application;
- 5) the address of the communication partner whose characteristics are to be queried. This address consists of the partner name and the processor name;
- 6) the length of the area in which DCAM is to enter the return information.

DCAM then returns:

- 3) the return code;
- 4) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff). In this case there is no further return information;
- 7) the partner characteristics (for interpretation of this information, see page 135).

3.2.2.3 Display terminal and character sets (MONCHARS)



This section applies only to DCAM(NEA) transport service applications.

CALL	USING	Parameter input		Description
		NEA	ISO	
YINQUIRE	→77 FUNKTION	"MCS"		1)
	→01 APP-NAME. 02 APPNAME	application-name		2)
	02 FDBK SYN- CHRONIZED.			
	03 RCD	return-code		3)
	03 ECD	error-code		4)
	03 IND			
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME	partner-name processor-name		5) 5)
02 .				
→01 MCS-BEREICH 02 MCS-LAENGE 02 MCS-DATEN COPY VTSUMONC	≥14 ≤ 30		6) 7)	

Information about display terminal and character sets

Key:

FUNKTION = function
 BEREICH = area
 LAENGE = length
 DATEN = data

Mandatory specifications:

- 1) the function of the call;
- 2) the name of the DCAM application;
- 5) the address of the communication partner whose characteristics are to be queried. This address consists of the partner name and the processor name;
- 6) the length of the area in which DCAM is to enter the return information.
Length = 14 if no character set is used.

DCAM then returns:

- 3) the return code;
- 4) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff). In this case there is no further return information;
- 7) the information about display terminals and character sets (see the chapter entitled "Status information" in the "VTSU User Guide").

3.2.2.4 Basic terminal information (BTERMINF)



This section applies only to DCAM(NEA) transport service applications.

CALL	USING	Parameter input		Description
		NEA	ISO	
YINQUIRE	→77 FUNKTION	"BTI"		1)
	→01 APP-NAME. 02 APPNAME	application-name		2)
	02 FDBK SYN- CHRONIZED.			
	03 RCD	return-code		3)
	03 ECD	error-code		4)
	03 IND			
	→01 CONN-NAME. 02 PTNNAME	partner-name		5)
	02 PRONAME	processor-name		5)
	.			
	→01 BTI-BEREICH 02 BTI-LAENGE	24		6)
02 BTI-DATEN COPY VTSUBAS			7)	

Basic terminal information

Key:

FUNKTION = function
 BEREICH = area
 LAENGE = length
 DATEN = data

Mandatory specifications:

- 1) the function of the call;
- 2) the name of the DCAM application;.
- 5) the address of the communication partner whose characteristics are to be queried. This address consists of the partner name and the processor name;
- 6) the length of the area in which DCAM is to enter the return information.

DCAM then returns:

- 3) the return code;
- 4) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff). In this case there is no further return information;
- 7) basic information about the terminal (see the chapter entitled "Status information" in the "VTSU User Guide").

3.2.2.5 Peripherals (PEROTERM)



This section applies only to DCAM(NEA) transport service applications.

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 10px;">YINQUIRE</div>	→77 FUNKTION	"POT"		1)
	→01 APP-NAME. 02 APPNAME	application-name		2)
	02 FDBK SYN- CHRONIZED.			
	03 RCD	return-code		3)
	03 ECD	error-code		4)
	03 IND			
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME	partner-name processor-name		5) 5)
02 .				
→01 POT-BEREICH 02 POT-LAENGE 02 POT-DATEN COPY VTSUPERP	8		6) 7)	

Information about peripherals

Key:

FUNKTION	=	function
BEREICH	=	area
LAENGE	=	length
DATEN	=	data

Mandatory specifications:

- 1) the function of the call;
- 2) the name of the DCAM application.
- 5) the address of the communication partner whose characteristics are to be queried. This address consists of the partner name and the processor name;
- 6) the length of the area in which DCAM is to enter the return information.

DCAM then returns:

- 3) the return code;
- 4) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff). In this case there is no further return information;
- 7) information about the peripherals (see the chapter entitled "Status information" in the "VTSU User Guide").

3.2.2.6 Partner count (COUNTPTN)

CALL	USING	Parameter input		Description
		NEA	ISO	
YINQUIRE	→77 FUNKTION		"CNT"	1)
	→01 APP-NAME. 02 APPNAME .		application-name	2)
	02 FDBK SYN- CHRONIZED. 03 RCD		return-code	3)
	03 ECD		error-code	4)
	03 IND			
	→01 CONN-NAME. . . .		the connection structure is required for the sake of compatibility; entries are not interpreted.	
	→01 CNT-BEREICH. 02 CNT-LAENGE 02 ANZ-VERBINDUNGEN 02 ANZ-AUFFORDERUNGEN		4	5)
		number-of-connections	6)	
		number-of-requests	7)	

Partner count

Key:

FUNKTION = function
 BEREICH = area
 LAENGE = length
 ANZ-VERBINDUNGEN = connection count
 ANZ-AUFFORDERUNGEN = request count

Mandatory specifications:

- 1) the function of the call;
- 2) the name of the DCAM application.
- 5) the length of the area in which DCAM is to enter the return information.

DCAM then returns:

- 3) the return code;
- 4) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff). In this case there is no further return information;
- 6) the number of partners currently connected to this application;
- 7) the number of connection requests currently waiting for acceptance by this application.

3.2.3 Rejecting a connection request

CALL	USING	Parameter input		Description
		NEA	ISO	
YREJLOG	→01 APP-NAME.			
	02 APPNAME		application-name	1)
	.			
	02 FDBK SYN- CHRONIZED.			
	03 RCD		return-code	2)
	03 ECD		error-code	3)
	03 IND			
	→01 CONN-NAME.			
	02 PTNNAME		partner-name	4)
	02 PRONAME		processor-name	4)
	.			

Rejecting a connection request

Mandatory specifications:

- 1) the name of the DCAM application;
- 4) the partner name and processor name.

Note

YREJLOG must be used in COBOL together with YINQUIRE. A YINQUIRE (TOP) enters the partner name and processor name in the PTNNAME and PRONAME items and therefore always needs to be specified beforehand.

DCAM then returns:

- 2) the return code;
- 3) the error code if the call could not be executed or an error occurred during execution.

3.2.4 Changing the characteristics of a connection

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; display: inline-block; padding: 2px; margin-bottom: 5px;">YCHANGE</div>	→01 APP-NAME. 02 APPNAME . 02 FDBK SYN- CHRONIZED. 03 RCD 03 ECD 03 IND	application-name return-code error-code		1) 2) 3)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME . 02 EDITIN. 03 TRANSF 03 GETBS 03 GETFC 03 LCASE 02 EDITOUT. 03 TRANSF 03 HCOPY 03 HOM 02 EDIT 02 PROC 03 TRUNC 03 SYSCODE →01 VTLG-NAME. 02 CODEIDEN . 02 CODEGRNO 02 CODEGROUP 03 CODENAM 03 CODENO 03 CODEVAL	partner-name processor-name { "PHY" "FOR" } "YES" "YES" "YES" { "PHY" "FOR" } "YES" "YES" "YES" "NO" "NO" structure-name (0<quantity<9) group-name (0<amount<9) codes		4) 4) 5) 6) 7) 8) 9) 10) 11) 12) 13) 14) 15) 16) 17) 18)

Changing the characteristics of a connection

Mandatory specifications:

- 1) the name of the DCAM application;
- 4) the address of the communication partner whose connection characteristics are to be changed. This address consists of the partner name and processor name.

All further items in the connection structure (input, output and processor functions) are interpreted by DCAM. As an option, entries can be changed, i.e. reset with new values. The following description of the items is only brief because a detailed description of the connection was given on page 73ff.

The following can be changed:

- 5) input message editing in either "PHY" (editing by the user) or "FOR" (use of format handling) (NEA transport service only). If this entry is omitted, "LIN" is assumed (use of the line terminal);
- 6) transmission of backspace characters in "YES" (NEA transport service only). If this entry is omitted, these characters are to be processed;
- 7) handling of the logical function key code of a terminal in "YES" (NEA transport service only). The function key code is transmitted as the first character of the message. If this entry is omitted, the function key code is not transmitted;
- 8) transmission of lowercase letters in "YES" (NEA transport service only). If this entry is omitted, lowercase letters are converted to upper case;
- 9) output message editing in either "PHY" (editing by the user), or "FOR" (use of format handling) (NEA transport service only). If this entry is omitted, "LIN" is assumed (use of the line terminal);
- 10) the simultaneous printout of the message on the attached printer in "YES" (NEA transport service only). If this entry is omitted, printout is suppressed;
- 11) the treatment of logical lines in "YES"; the message is output unstructured (NEA transport service only). If this entry is omitted, message output is structured.

Changes 5) through 10) are, however, only possible if message editing by the system (EDIT=SYSTEM) was specified when setting up the connection.

Further **options** are:

- 12) handling of excess length messages in "NO"; the excess portion is indicated and kept for a further YRECEIVE. If this entry is omitted, the excess portion is truncated and discarded;
- 13) code conversion in "NO"; bit patterns are transmitted (NEA transport service only). If this entry is omitted, conversion to EBCDIC is required (only meaningful if EDIT="USR" has been specified for YOPNCON).

For DCAM(NEA) transport service applications, if messages are distributed using distribution codes, some values in the distribution structure may be altered.

Mandatory specification:

- 14) the symbolic name of the distribution structure as given for YOPNCON (NEA transport service only).

The following can be changed:

- 15) the number of distribution code groups connected (NEA transport service only);
- 16) the symbolic name of the distribution code group(s) (NEA transport service only). Care must be taken that the symbolic name of the distribution code group(s) is unique. If another connection has used the same distribution structure for YOPNCON, then a new name must be chosen when the entry is changed;
- 17) the number of distribution codes in the code group (NEA transport service only);
- 18) the distribution codes themselves (NEA transport service only).

DCAM then returns:

- 2) the return code;
- 3) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.2.5 Canceling a request

A connection request can be canceled only if the partner has not yet accepted it.

CALL	USING	Parameter input		Description
		NEA	ISO	
YCLSCON	→01 APP-NAME.			
	02 APPNAME		application-name	1)
	.			
	02 FDBK SYN- CHRONIZED.			
	03 RCD		return-code	2)
	03 ECD		error-code	3)
	03 IND			
	→01 CONN-NAME.			
	02 PTNNAME		partner-name	4)
	02 PRONAME		processor-name	4)
.				

Canceling a request

Mandatory specifications:

- 1) the name of the DCAM application;
- 4) the address of the communication partner, consisting of the partner name and the processor name.

DCAM then returns:

- 2) the return code;
- 3) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.2.6 Clearing down a connection

A connection established via request and acceptance can be cleared down again using the YCLSCON call. Data not yet picked up will be lost. Entries and return information are as described under "Canceling a request" (see above).

3.3 Data communication-oriented CALLS

After a connection has been successfully established, the communication partners are in a position to transmit data.

A send or receive call always implies the transfer of data from the user area to the data area of the communication system and vice versa. Thus, for instance, a call to transmit data is considered successfully completed as soon as the data has been transferred to the communication system data area. In DCAM(NEA) transport service applications, the control functions for distribution code assignment are also part of the data transmission function.

The following calls are available:

- **YSEND** Send a message.
- **YRECEIVE** Receive a message.
- **YRESET** Reset the CS/CA state of a connection.



and additionally for DCAM(NEA) transport service applications:

- **YRECEIVE** Receive a transport acknowledgment
- **YPERMIT** Permit a task of the task group to receive messages with certain distribution codes.
- **YFORBID** Forbid a task of the task group from receiving messages with certain distribution codes.

The two calls for distribution code control can only be issued by a primary task.

3.3.1 Sending a message

CALL	USING	Parameter input		Description
		NEA	ISO	
<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">YSEND</div> <div style="display: flex; align-items: center;"> <div style="border-left: 1px solid black; border-right: 1px solid black; border-bottom: 1px solid black; width: 10px; height: 100px; margin-right: 5px;"></div> <div style="margin-left: 5px;"> <p>→01 APP-NAME. 02 APPNAME</p> <p>→01 CONN-NAME. 02 PTNNAME 02 PRONAME</p> <p>→01 BEF-NAME. 02 TOVAL 02 SEQNO 02 OPTCD. 03 SPEC 03 CS</p> <p>03 DG 03 NORMAL 03 TRUNC 03 TACK 03 BELL 03 FHS</p> <p>02 FDB SYN- CHRONIZED.</p> <p>03 FDBK. 04 RCD 04 ECD 04 IND</p> <p>03 VTSUCB</p> <p>→01 VTSUCB</p> <p>→01 BEREICH1. 02 LAENGE1 02 TEXT1</p> <p>----- →01 FHS-MAIN-PAR</p> <p>----- →01 BEREICH2. 02 LAENGE2 02 TEXT2</p> </div> </div>	<p>application-name</p> <p>partner-name processor-name</p> <p>(0<number<65535)</p> <p>"YES"</p> <p>{ "ELE" "SUB" } "NO"</p> <p>"YES" "YES" "YES"</p> <p>return-code error-code</p> <p>"YES"</p> <p>{ (1≤n≤8) (1≤n≤maxln) } text</p> <p>see FHS User Guide</p> <p>length 1) number of bytes in the length</p>	<p>1)</p> <p>2) 2)</p> <p>3)</p> <p>4)</p> <p>5) 6)</p> <p>7) 8) 9)</p> <p>10) 11)</p> <p>12)</p> <p>13) 14)</p> <p>15)</p> <p>16) 17)</p>		

1) The length depends on whether restart, partial formatting or both are being used.

Sending a message

Key:

BEREICH = area
LAENGE = length

Mandatory specifications:

- 1) the name of the DCAM application;
- 2) the address of the communication partner, consisting of the partner name and the processor name;
- 13) the length of the message in the transmit area. The message length can be between 1 character and the maximum length returned for YOPNCON;

Note for DCAM(NEA) transport service applications

Express messages can be between 1 and 8 characters long.

The length need not be specified where FHS="YES". However, the length item must be present; for details see "FHS User Guide";

- 14) the text of the message to be sent. This item must be at least as long as the length given in 12). For DCAM(NEA) transport service applications, where FHS="YES", the FHS transfer area is to used (for details see the "FHS User Guide").

Optional specifications:

- 3) the sequence number to be passed to the recipient with the message (NEA transport service only). This number also serves to identify the transport acknowledgment for the message and is returned with the acknowledgment. A value between 1 and 65535 can be entered as a sequence number. The default value is zero;
- 4) that the originator-oriented queue (CS state of the connection) is to be used for further messages coming from this partner. For DCAM(NEA) transport service applications, this entry is only meaningful if messages are not distributed using distribution codes;
- 5) whether, for DCAM(ISO), the transmitted data unit is the last of a message group ("GROUP"), or whether further message units are to follow ("ELE"). This item is interpreted only if the connection has been opened with MDATA="Y";

if, for DCAM(NEA), the attributes EDIT="USR" or EDIT="SYS" and TRANSF="PHY" have been specified for YOPNCON; this entry indicates which element of a message is being transmitted:

"ELE" element of a group or subgroup;

"SUB" last element of a subgroup.

If this entry is omitted, the message is sent as the last or only element (GROUP).

This entry is transmitted with the message and passed to the recipient. If the latter is a terminal, the information is converted, as far as possible, into control information (block, last block of a message, last block of the transmission);

- 6) that this is to be an express message (NEA transport service only). It is transmitted to the partner, bypassing all data flow control and overtaking messages sent previously but not yet picked up. The maximum length of such a message is 8 bytes. For this function, the connection must have been set up with the attribute EDIT="USR". If this entry is omitted, the message is sent as a normal message;
- 7) that a positive acknowledgment for transport of the message, including transfer to the recipient, is expected (NEA transport service only). If this entry is omitted, no positive acknowledgment is expected. For this function, the relevant application must accept acknowledgments (TACK="PRI", TACK="REQ" for YOPEN);
- 8) that an audible alarm be activated (NEA transport service only). For this function, the connection must have been established with the attribute EDIT="SYS";
- 9) processing with the integrated Format Handling System FHS (NEA transport service only). If FHS is not used, formatted data must be provided in the transmit area. For this function, EDIT="SYS" and TRANSF="FOR" must have been specified for the connection;
- 12) that VTSUCB is to be used for message editing (NEA transport service only), (see chapter entitled "VTSU control block" in the "VTSU User Guide");
- 15) the name of an FHS operand structure, if formatting was performed by FHS (FHS="YES") (NEA transport service only);
- 16) the length of an administrative area for FHS for partial formatting and restart (NEA transport service only);
- 17) reserved for FHS (NEA transport service only); for the minimum length, see the "FHS User Guide".

Note for DCAM(NEA) transport service applications

If DCAMVER = 0800 was specified (i.e. DCAM version \geq 8.0), when FHS is used for output formatting the FHS operand area (FHS-MAIN-PAR) and, if specified, the FHS administrative area (BEREICH2) are copied internally by DCUS and buffered until the next YRECEIVE from the same partner. These copies are made available to FHS for input formatting. After input formatting by FHS the copies are returned to the user again in FHS-MAIN-PAR and BEREICH2.

DCAM then returns:

- 10) the return code;
- 11) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.3.2 Receiving a message or transport acknowledgment

Access to the originator-oriented queue or the common receiver queue requires different specifications. The receive call is executed asynchronously if SYN="NO" is specified. The default value is SYN="YES".



A receive call is issued as an unspecific call. The user recognizes from the feedback information whether he or she has received a message or a transport acknowledgment. Access to the originator-oriented queue or the common receiver queue requires a prior assignment (page 120).

3.3.2.1 Accessing the common receiver queue

For this function, this queue must have already been set up. This can be done with YOPNCON, YSEND and YRECEIVE.



This condition does not apply when using distribution codes. Messages are received from any partner in accordance with prior assignment.

CALL	USING	Parameter input		Description
		NEA	ISO	
YRECEIVE	→01 APP-NAME. 02 APPNAME		application-name	1)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME		partner-name processor-name	2) 2)
	→01 BEF-NAME. 02 TOVAL 02 SEQNO 02 OPTCD. 03 SPEC 03 CS 03 Q		(0<n<32768) "YES" "YES" "YES"	3) 4) 5) 6)
	03 TRUNC 03 TACK		{"TRC" "KEE"}	7)
	03 FHS 02 FDB SYN- CHRONIZED. 03 ASEQNO 03 TACKNO	"YES"	sequence-number acknowledgment- number	8) 9) 10)
	03 ARECLN 03 FDBK. 04 RCD 04 ECD 04 IND		length return-code error-code indicator	11) 12) 13) 14)
	03 SYN 03 VTSUCB	"YES"	"NO"	15) 16)
	→01 VTSUCB			
	→01 BEREICH1. 02 LAENGE1		(8≤n≤maxln) length	17) 18)
	02 TEXT1		text	19)
	→01 FHS-MAIN-PAR .		see FHS User Guide	20)
	→01 BEREICH2. 02 LAENGE2 02 TEXT2		length 1) number of bytes in the length	21) 22)

1) the length depends on whether restart, partial formatting or both are being used.

Accessing the common receiver queue

Key:

BEREICH = area
LAENGE = length

Mandatory specifications:

- 1) the name of the DCAM application;
- 17) the length of the area in which DCAM is to enter the message.

Where FHS="YES", the length itself need not be entered but the length item must be present (for details see "FHS User Guide").

Optional specifications:

- 3) the wait time for this call, if it is to be queued. The wait time must be at least 1 second and at most 32767 seconds. The default value is 600 seconds;
- 5) that the originator-oriented queue (CS state of the connection) is to be used for future calls. All subsequent messages of the partner will be transferred to that task. If, in the event of message overflow, the excess portion is to be picked up by another YRECEIVE, then this queue has to be used. This entry holds for all subsequent YRECEIVE calls until it is changed again.

It is not required when distribution codes are being used and is not interpreted. Any transport acknowledgment received will not be interpreted either;
- 6) that the call is to be queued if it cannot be executed immediately because no message is pending. If this entry is omitted, the call is terminated immediately and must be repeated later if necessary;
- 7) that in the event of an excessively long message, the excess portion be held for a further YRECEIVE ("KEE"=KEEP). For this function, the CS state must have been set for the connection as explained in 5). The entry "TRC"=TRUNC means that the rest of the message is discarded. If this entry is omitted, the entries made in the connection structure for YOPNCON are assumed;
- 8) that the integrated Format Handling System FHS is used (NEA transport service only). An FHS parameter structure must already have been completely filled with values in YSEND (i.e. all entries for input and output formatting must be input). The FHS parameter structure is buffered by DCUS internally until the next YRECEIVE. Nothing need be specified here in the FHS parameter structure. Its purpose is to return information through DCAM;

- 15) whether the call is to be executed asynchronously (SYN="NO"). For further details, see the YWAIT call, page 123ff;
- 16) that VTSUCB is to be used for message editing (NEA transport service only) (see chapter entitled "VTSU control block" in the "VTSU User Guide");
- 20) the name of an FHS operand structure when formatting is carried out by FHS (NEA transport service only);
- 21) the name of an administrative area for FHS (NEA transport service only);
- 22) the length of the administrative area (NEA transport service only).

Note for DCAM(NEA) transport service applications

If the data is to be edited using FHS (FHS="YES"), an FHS operand structure must already have been completely filled with values for input and output formatting in YSEND. It is buffered by DCUS internally until the next YRECEIVE on the same connection.

FHS-MAIN-PAR and BEREICH2 are evaluated depending on the DCAMVER operand of YOPEN:

- where DCAMVER is not specified (DCAMVER < V8.0):
FHS-MAIN-PAR is not specified for YRECEIVE. Input formatting is performed with the FHS-MAIN-PAR of the preceding YSEND. The feedback information from FHS is in the FHS-MAIN-PAR of the preceding YSEND. If there are several connections with different formats, YRECEIVE (ANY) is not possible.
The FHS operation is not possible with applications where SHARE="YES".
- where DCAMVER = 0800 (i.e. DCAM version is \geq V8.0):
FHS-MAIN-PAR must always be specified. The feedback information is made available in FHS-MAIN-PAR.
BEREICH2 need only be specified for formatting with restart or partial formatting (see also notes on YSEND).

DCAM then returns:

- 2) the address of the originator of the message or transport acknowledgment. The address consists of the partner name and processor name;
- 9) the sequence number of the message as assigned by the partner (NEA transport service only). For terminals, it is generated in the communication system;
- 10) if a transport acknowledgment has been received, the sequence number of the acknowledged message (NEA transport service only);

11) and 18)
the actual length of the received message, or the length of the excess portion, if the message was longer than specified in the EMPF-LAENGE (receive length) item;

19) the message itself.

If FHS="YES" was specified, the FHS transfer area is used (for details see the "FHS User Guide"),

and also:

12) the return code;

13) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff);

14) the indicator; this indicates whether further data units are to follow.

Note for DCAM(NEA) transport service applications

The same FHS structure is used for YRECEIVE as was used for the last associated YSEND.

After the return code "partial formatting not completed" has appeared in the instruction structure, YRECEIVE must be repeated until the return code "partial formatting completed" appears so that the remaining partial formats can be picked up.

3.3.2.2 Accessing the originator-oriented queue

For this function, this queue must have already been set up. This can be done during YOPNCON, YSEND and YRECEIVE. For DCAM(NEA) transport service applications this condition is not applicable when using distribution codes.

CALL	USING	Parameter input		Description
		NEA	ISO	
YRECEIVE	→01 APP-NAME. 02 APPNAME . .		application-name	1)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME . .		partner-name processor-name	2) 2)
	→01 BEF-NAME. 02 TOVAL 02 SEQNO 02 OPTCD. 03 SPEC 03 CS 03 Q . 03 TRUNC 03 TACK .		(0 < n < 32768) "YES" "YES" "YES" { "TRC" "KEE" }	3) 4) 5) 6) 7)
	02 FDB SYN- CHRONIZED. 03 ASEQNO 03 TACKNO	sequence-number acknowledgment- number		8) 9)
	03 ARECLN 03 FDBK. 04 RCD 04 ECD 04 IND . 03 SYN	length return-code error-code indicator		10) 11) 12) 13) 14)
	→01 EMPF-BEREICH. 02 EMPF-LAENGE . 02 EMPF-TEXT		"NO" (8 ≤ n ≤ maxln) length text	15) 16) 17)

Accessing the originator-oriented queue

Key:

EMPF-BEREICH= receive area
EMPF-LAENGE = receive length
EMPF-TEXT = receive text

Mandatory specifications:

- 1) the name of the DCAM application;
- 2) the address of the partner from whom the message is expected. This address consists of the partner name and processor name;
- 4) that the message is to be received from a specific partner;
- 15) the length of the area in which DCAM is to enter the message.

The specifications already described for accessing the common receiver queue (page 112) are **optional**.

DCAM returns the information listed on page 112ff with the exception of 2).

3.3.3 Combined sending and receiving

Combined sending and receiving is not possible in COBOL as YSENDREC is not available.

3.3.4 Changing the CS/CA state of a connection

CALL	USING	Parameter input		Description
		NEA	ISO	
YRESET	→01 APP-NAME. 02 APPNAME .		application-name	1)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME .		partner-name processor-name	2) 2)
	→01 BEF-NAME. .			
	02 OPTCD. 03 SPEC 03 CS .		"YES" "YES"	3) 4)
	03 FDBK. 04 RCD 04 ECD 04 IND		return-code error-code indicator	5) 6) 7)

Changing the CS/CA state of a connection

Mandatory specifications:

- 1) the name of the DCAM application;
- 2) the address of the communication partner, consisting of partner name and processor name;
- 3) that the call relates to a particular partner.

Optional specification:

- 4) that the originator-oriented queue is to be used (CS). If this entry is omitted, the common receiver queue is used (CA).

Note

YRESET can only be used with SPEC="YES" and, additionally for DCAM(NEA) transport service applications, DISCO="NO"

3.3.5 Controlling distribution code assignment

 This section applies only to DCAM(NEA) transport service applications.

Two calls are available to the primary task for controlling the assignment of distribution codes. One call (YPERMIT) assigns distribution code names to distribution code groups, while the other call (YFORBID) cancels such assignments without making any new ones.

3.3.5.1 Assigning distribution code names to distribution code groups

 This section applies only to DCAM(NEA) transport service applications.

CALL	USING	Parameter input		Description
		NEA	ISO	
YPERMIT	→01 APP-NAME. 02 APPNAME	application-name		1)
	02 FDBK SYN- CHRONIZED.			
	03 RCD	return-code		2)
	03 ECD	error-code		3)
	03 IND			
	→77 VT-NAME	distribution-name		4)
	→01 VTLG-NAME. 02 CODEIDEN	structure-name		5)
	02 CODEGRNO	(0<quantity<9)		6)
	02 CODEGROUP			
	03 CODENAM	group-name		7)
03 CODENO	(0<amount<9)		8)	
03 CODEVAL	codes		9)	
→77 CG-NAME	group-name		10)	

Assigning distribution code names to distribution code groups

Mandatory specifications:

- 1) the name of the DCAM application;
- 4) the distribution code name to be assigned to a distribution code group;
- 5) the symbolic name of the distribution structure containing the distribution code group to which the assignment is to be made;
- 6) the number of distribution code groups in that structure;
- 7) and 10) the symbolic name of the distribution code group to which the assignment is to be made.

Optional specifications:

- 8) the number of codes contained in the group concerned;
- 9) the values of the distribution codes.

Note

Specifications 8) and 9) are only interpreted if the structure has not already been used by YOPNCON for establishing a connection or by another YPERMIT. Changes should only be made with YCHANGE (page 103).

DCAM then returns:

- 2) the return code;
- 3) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.3.5.2 Canceling the assignment



This section applies only to DCAM(NEA) transport service applications.

CALL	USING	Parameter input		Description
		NEA	ISO	
YFORBID	→01 APP-NAME .	application-name		1)
	02 APPNAME			
	02 FDBK SYN- CHRONIZED.			
	03 RCD	return-code		2)
	03 ECD	error-code		3)
	03 IND			
	→77 VT-NAME	distribution-name		4)

Canceling the assignment

Mandatory specifications:

- 1) the name of the DCAM application;
- 4) the distribution code name whose assignment to a distribution code group is to be canceled. The name can be 1-8 characters long. It must be left-justified, and, if necessary, the line must be filled with blanks to the right.

Note

This call is only available to the primary task. The application must be shareable and DISCO="YES" must be set.

DCAM then returns:

- 2) the return code;
- 3) the error code if the call could not be executed or an error occurred during execution (for error codes see page 177ff).

3.4 CALLs for asynchronous processing

Instructions to DCAM can be executed in two ways:

- **Synchronous execution:** simple to use, few specifications, but wait times are not used in the optimum way
- **Asynchronous execution:** optimum utilization of wait times, but the program is more complex (only with YOPNCON and YRECEIVE).

Synchronous execution

With synchronous execution (SYN="YES" is the default value in the instruction structure), control is only returned to the program after the instruction has been executed.

But what happens if the call cannot be executed immediately, e.g. with YOPNCON or YRECEIVE? In this case the user can either wait for the event (e.g. the arrival of the message) to occur or have control returned immediately by means of the appropriate indication:

- **Q="YES"**

The call is entered **in a queue** until the expected event occurs. The maximum wait time can be specified with TOVAL (timeout value). The call is terminated, i.e. control is returned to the user, after the event has occurred or the wait time has elapsed.

- **Q="NO"**

The call is **terminated immediately** even if the expected event has not yet occurred. With a YOPNCON (acceptance of a request) this may mean that a connection request was not received from the partner. With a YRECEIVE the message has not yet arrived. In this case, the user may have to repeat the call later.

Note

Only one synchronous call which initiates a wait time can be executed at any given time.

Asynchronous execution

Instructions which may require the system to wait for an event to occur can also be executed asynchronously. This means that the **SYN** item must have the value "NO" which is only possible with

- **YOPNCON** (see page 72ff) and
- **YRECEIVE** (see page 107ff).

When the YWAIT call is issued, the system waits for an event to occur.

3.4.1 Waiting for termination of asynchronous DCAM jobs

After an asynchronous call (YOPNCON, YRECEIVE) an event should occur. The following events are possible:

- OPENED
- LETTER
- GOSIGNAL
- LOSCON
- NOEVENT

Event	Meaning
OPENED	The YOPNCON request has been completed; for details, see instruction structure and connection structure
LETTER	The YRECEIVE request has been completed; for details, see instruction structure and connection structure
GOSIGNAL	The memory space bottleneck has been cleared; for details, see instruction structure and connection structure
LOSCON	The connection was cleared down by the system or by the partner; for details, see instruction structure
NOEVENT	No event has occurred; for details, see instruction structure

The user receives the wait job's return code in the feedback information. If an event has occurred (RSI=0), the return code for completion of the asynchronous DCAM job or the LOSCON reason is contained in the instruction structure that was specified with the YWAIT job.

For the limit values for asynchronous calls, see page 176.

Note

When waiting for an event DCUS uses eventing internally. In this way a task-specific wait queue is implemented for DCAM events.

3.4.1.1 Receiving messages

After completion of a YRECEIVE request the DCAM event LETTER occurs. The remaining partial formats are picked up with another YRECEIVE.

CALL	USING	Parameter input		Description
		NEA	ISO	
YWAIT	→01 WAIT-NAME. 02 LIFETIM . 03 RETCODE 05 RRS 05 RSI 03 EREIGNIS .		wait-time	1)
			RS-return-code	2)
			SI-return-code	2)
			event	3)
	→01 APP-NAME. 02 APPNAME .		application-name	4)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME .		partner-name processor-name	5) 5)
	→01 BEF-NAME. 02 FDB SYN- CHRONIZED. 03 ASEQNO 03 TACKNO	sequence-number acknowledgment- number		6) 7)
	03 ARECLN 03 FDBK. 04 RCD 04 ECD 04 IND		length return-code error-code indicator	8) 9) 10) 11)
	- - - →01 FHS-MAIN-PAR. .	see FHS User Guide		12)
	L - - - →01 BEREICH2 02 LAENGE2 02 TEXT2	length number of bytes in the length		13) 14)

Receiving messages

Key:

- EREIGNIS = event
- BEREICH = area
- LAENGE = length

Optional specification:

- 1) the time, in seconds, which the task is to wait for the event to occur; the default value is 60.

DCAM then returns:

- 2) the return code of the event;
- 3) the event identifier LETTER;
- 4) the application name;
- 5) the address of the communication partner in the connection structure, consisting of the partner name and the name of the processor to which the partner is connected;
- 6) the sequence number which the partner has assigned to the message (NEA transport service only); in the case of terminals this is generated in the communication system;
- 7) the sequence number of the acknowledged message if a transport acknowledgment has been received (NEA transport service only);
- 8) the actual length of the received message or the length of the message remainder if the message was longer than was specified for EMPF=LAENGE (receive length) in YRECEIVE;
- 9) the return code;
- 10) the error code if the call could not be executed or an error occurred during execution;
- 11) the indicator;
- 12) the FHS operand structure if formatting is performed by FHS (NEA transport service only);
- 13) the length of the FHS administrative area (NEA transport service only);
- 14) the FHS administrative area (NEA transport service only).

Note for DCAM(NEA) transport service applications

After the return code "partial formatting not completed" has appeared in the instruction structure, YRECEIVE must be repeated until the return code "partial formatting completed" appears so that the remaining partial formats can be picked up.

3.4.1.2 Setting up a connection

After completion of a YOPNCON request the event OPENED occurs.

CALL	USING	Parameter input		Description
		NEA	ISO	
YWAIT	→01 WAIT-NAME. 02 LIFETIM . 03 RETCODE 05 RRS 05 RSI 03 EREIGNIS .		wait-time	1)
	→01 APP-NAME. 02 APPNAME .		application-name	2) 2) 3)
	→01 CONN-NAME. 02 PTNNAME 02 PRONAME .		partner-name processor-name	4) 5) 5)
	02 EDIT .	{ "USR" "SYS" }		6)
	03 APPSTART 02 MAXLN	{ "YES" "NO" }	maxln	7) 8)
	02 PTNCHAR. 03 PTNTYPE 03 DEVTYPE 03 CHARSET 03 DEVSEC		first four items of the partner characteristics	9)
	→01 BEF-NAME. .		Length of connection message	10)
	03 ARECLN			11)
	03 FDBK. 04 RCD		return-code	12)
	04 ECD		error-code	

Connection setup

Key:

EREIGNIS = event

Optional specification:

- 1) the time, in seconds, which the task is to wait for the event to occur; the default value is 60.

DCAM then returns:

- 2) the return code of the event;
- 3) the event identifier (OPENED);
- 4) the application name;
- 5) the address of the communication partner in the connection structure, consisting of the partner name and the name of the processor to which the partner is connected;
- 6) the type of message editing (NEA transport service only);
- 7) the start of data transmission (NEA transport service only);
- 8) the maximum message length;
- 9) the partner characteristics (see page 135) (NEA transport service only);
- 10) the actual length of the connection message or the length of the message remainder if the connection message was longer than was specified in BEREICH1 in YOPNCON;
- 11) the return code;
- 12) the error code if the call could not be executed or an error occurred during execution.

3.4.1.3 GO notification

The DCAM event GOSIGNAL occurs when a bottleneck has been cleared, if the connection was set up with SIGNAL = 'YES'.

CALL	USING	Parameter input		Description
		NEA	ISO	
YWAIT	→01 WAIT-NAME.			
	02 LIFETIM		wait-time	1)
	.			
	03 RETCODE			
	05 RRS		RS-return-code	2)
	05 RSI		SI-return-code	2)
	03 EREIGNIS		event	3)
	.			
	→01 APP-NAME.			
	02 APPNAME		application-name	4)
	.			
	→01 CONN-NAME.			
	02 PTNNAME		partner-name	5)
	02 PRONAME		processor-name	5)
	.			
	→01 BEF-NAME.			
.				
02 FDB SYN- CHRONIZED.				
.				
03 FDBK.				
04 RCD		return-code	6)	
04 ECD		error-code	7)	
04 IND		indicator	8)	

GO notification

Key:

EREIGNIS = event

Optional specification:

- 1) the time, in seconds, which the task is to wait for the event to occur; the default value is 60.

DCAM then returns:

- 2) the return code of the event;
- 3) the event item identifier GOSIGNAL;
- 4) the application name;
- 5) the address of the communication partner in the connection structure, consisting of the partner name and the name of the processor to which the partner is connected;
- 6) the return code;
- 7) the error code if the call could not be executed or an error occurred during execution;
- 8) the indicator.

3.4.1.4 Message on connection cleardown

After connection cleardown the DCAM event LOSCON occurs.

CALL	USING	Parameter input		Description
		NEA	ISO	
YWAIT	→01 WAIT-NAME.			
	02 LIFETIM		wait-time	1)
	.			
	03 RETCODE			
	05 RRS		RS-return-code	2)
	05 RSI		SI-return-code	2)
	03 EREIGNIS		event	3)
	.			
	→01 APP-NAME.			
	02 APPNAME		application-name	4)
	.			
	→01 CONN-NAME.			
	02 PTNNAME		partner-name	5)
	02 PRONAME		processor-name	5)
	.			
	→01 BEF-NAME.			
04 LOSCON-REASON		Reason for cleardown	6)	
.				

Message on connection cleardown

Key:

EREIGNIS = event

Optional specification:

- 1) the time, in seconds, which the task is to wait for the event to occur; the default value is 60.

DCAM then returns:

- 2) the return code of the event;
- 3) the event item identifier LOSCON;
- 4) the application name;
- 5) the address of the communication partner in the connection structure, consisting of the partner name and the name of the processor to which partner is connected;
- 6) the reason for connection clear-down (LOSCON-REASON item in the instruction structure).

Reasons for a connection cleardown (LOSCON)

Value	ISO	NEA	Reason
0	x	x	Connection termination was requested by the user.
4		x	Invalid EDIT combination.
8	x	x	Connection cleardown due to processing error at the partner terminal.
12	x	x	The partner is not available (e.g. terminal failure).
16	x	x	Connection termination was initiated by the system operator.
20	x	x	The partner has lost his connection. He will make another request for connection setup.
24	x	x	Connection termination is due to inconsistency or an error in the data transmission network.
32	x	x	Warning: connection termination.
36		x	Partner characteristic is not accepted by the partner.
40	x	x	Connection cleardown after request by system administration.
44		x	Connection cleardown due to error in the user service connection protocol.
52		x	Connection cleardown due to error in the user service protocol.
56	x	x	Connection cleardown due to error in the transport system.
64		x	Connection cleardown due to syntax error in the user message of the connection protocol (LOGON message).
84		x	Network priority not accepted by the partner.
88		x	Resource bottleneck in the partner system (too many NTACKs).
92		x	VTSU of the partner cannot be activated.
96	x	x	Connection cleardown due to protocol error of the communication partner.
100		x	Fatal UCON error
104		x	Application name invalid: first character not \$

Value	ISO	NEA	Reason
108		x	Invalid processor (not own processor)
112		x	Could not create task for verification
116		x	Invalid CID for this OP-ID
120		x	Internal UCON error (authentication not possible)
124		x	No entry available in ECRNAM table
128		x	DCAM version is less than 10
132		x	Not a chipcard terminal
136		x	Protocol inconsistent
140		x	Chipcard subsystem not available
144		x	Error in KVP protocol

4 Partner characteristics



This chapter applies only to DCAM(NEA) transport service applications

This chapter describes the status information on the partner characteristics for DCAM COBOL applications. VTSU supplies additional information on the terminal and the character sets, the peripherals and the basic terminal information. These topics are discussed in detail in the chapter entitled "COBOL interface" of the "VTSU User Guide".

Forms and contents of the item used by DCAM after a YINQUIRE call ("PTN" variant):

Level numbers, record names, and data names	PICTURE clause	Meaning
01 PTN-BEREICH.		
02 PTN-LAENGE	PIC 9(4) COMP.	Item length (user-specified: 26)
02 PTNCHAR 1.		
03 PTNTYPE	PIC 9(4) COMP.	Type of partner (DCAM application or terminal)
03 DEVTYPE	PIC 9(4) COMP.	Device type
03 CHARSET	PIC 9(4) COMP.	Available character set
03 DEVSEC	PIC 9(4) COMP.	Optional features of terminal
02 PTNCHAR 2.		
03 PHYS-EIGENSCHAFT.		Physical device properties
04 ZEILENLAENGE-P	PIC 9(4) COMP.	Line length
04 ZEILENANZAHL-P	PIC 9(4) COMP.	Number of lines
04 MAXIMALPUFFER-P	PIC 9(4) COMP.	Maximum device buffer size
03 LINE-EIGENSCHAFT.		Line terminal properties
04 ZEILENLAENGE-L	PIC 9(4) COMP.	Line length
04 ZEILENANZAHL-L	PIC 9(4) COMP.	Number of lines
04 MAXIMALPUFFER-L	PIC 9(4) COMP.	Maximum buffer size
03 LOGDAT	PIC 9(4) COMP.	Message editing capability
03 EDITIN	PIC 9(4) COMP.	Editing set for input
03 EDITOUT	PIC 9(4) COMP.	Editing set for output

Key:

BEREICH = area
 LAENGE = length
 EIGENSCHAFT = characteristics
 ZEILENLAENGE = line length
 ZEILENANZAHL = line quantity
 MAXIMAL = maximum
 PUFFER = buffer

The possible entries in the individual items are described in the following table.

Data name	Entry by DCAM	Meaning
PTNTYPE	0	Partner is an application
	1	Partner is a terminal
DEVTYPE	3	X.29 display terminal (similar to 8150)
	4	8150 Data Display Terminal
	8	Intelligent partner
	21	8151 Data Display Terminal
	22	8152 Data Display Terminal
	23	8110 Printer Terminal
	28	8121 Printer
	29	Printer Terminal PT80
	30	T1000 Teleprinter
	35	T100 Teleprinter
	38	FS100-E Teleprinter
	42	90037 Integrated Printer Terminal Controller
	43	8122 Printer
	44	8162 Data Display Terminal
	45	8160 Data Display Terminal
	46	8124 Printer
	48	APS Program
	51	X.29 printer terminal
53	9750 Data Display Terminal	
54	9003 Printer	
57	9770 Terminal (similar to 8151)	
59	9002 Printer	

Partner characteristics

Data name	Entry by DCAM	Meaning
DEVTYPE	61	3974 Display Unit
	63	9751 Data Display Terminal (similar to 8160)
	64	9752 Data Display Terminal (similar to 9750)
	65	9753 Personal Computer (similar to 8160)
	66	9001 Printer
	67	9731 Graphics Workstation
	69	9004 Printer
	76	9754 Data Display Terminal
	77	9749 Data Display Terminal (similar to 9750)
	78	9755 Data Display Terminal
	79	9763 Data Display Terminal
	85	VTX Terminal T-3000
	86	VTX Editing Terminal
	87	VTX Retrieval Terminal
	90	UTC for Teletex
	91	9012 Printer
	92	9013 Printer
	101	9001-31 Printer
	102	9001-8931 Printer
	104	9022 Printer
107	9011-18 Printer	
108	9011-19 Printer	
112	9021 Printer	

Data name	Entry by DCAM	Meaning
ZEILENLAENGE-P	a	a is the maximum number of characters that can be displayed in one line on the terminal
ZEILENANZAHL-P	b	b is the maximum number of lines that can be displayed on the screen of a data display terminal
MAXIMALPUFFER-P	c	c is the maximum size, in bytes, of the terminal's buffer
ZEILENLAENGE-L	d	d is the maximum number of characters that can be output by the line terminal in one line (effective only with printers)
ZEILENANZAHL-L	e	e is the maximum number of lines that can be output by the line terminal on one page in a single input/output operation
MAXIMALPUFFER-L	f	f is the maximum size in bytes of the line terminal's buffer
LOGDAT	11	printer terminal
	75	data display terminal
EDITIN	8192	LINE
	9216	LINE + GETBS
	12288	LINE + LCASE
	13312	LINE + GETBS + LCASE
	24576	PHYS
	53760	FORM
	54784	FORM + GETBS

Partner characteristics

Data name	Entry by DCAM	Meaning
EDITOUT	1024	LINE
	16896	FORM
	17921	PHYS
	33792	LINE + HCOPY
	50689	PHYS + HCOPY

Key:

ZEILENLAENGE = line length
ZEILENANZAHL = line quantity
MAXIMAL = maximum
PUFFER = buffer

5 Examples

5.1 DCAM(ISO) transport service

The following sample program implements a simple DCAM(ISO) transport service application:

```

ID DIVISION.
PROGRAM-ID. MIRROR.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.

        COPY YDDCUAPL.
        COPY YDDCUCON.
        COPY YDDUCOM.
        COPY YDDCUWAI.

01  IOAREA.
    02  LENGTH          PIC 9(4) COMP VALUE IS 0.
    02  TXT              PIC X(4096) VALUE IS SPACES.

01  DCAMVER            PIC 9(4) COMP-3 VALUE IS 0800.

PROCEDURE DIVISION.

*****
*   OPEN "MIRROR" APPLICATION   *
*****
        MOVE "Y" TO ISO IN APP-NAME.
        MOVE "NO" TO TACK IN APP-NAME.
        MOVE "MIRROR" TO APPNAME IN APP-NAME.
        CALL "YOPEN" USING APP-NAME DCAMVER.
        IF RCD IN APP-NAME NOT EQUAL 0
        THEN GO TO END.

```

```

*****
*   ESTABLISH CONNECTION (ACCEPT ANY)           *
*****
MOVE "YES" TO SYN IN BEF-NAME.
MOVE "YES" TO Q IN BEF-NAME.
MOVE "YES" TO SIGNAL IN CONN-NAME.
MOVE "N" TO MDATA IN CONN-NAME.
MOVE 900 TO TOVAL IN BEF-NAME.
CALL "YOPNCON" USING APP-NAME CONN-NAME BEF-NAME.
IF RCD IN BEF-NAME NOT EQUAL 0
THEN GO TO END.

*****
*   ISSUE YRECEIVE                             *
*****
PERFORM RECEIVE-ASY.

WAIT-FOR-EVENT.
*****
*   WAIT FOR EVENT                             *
*****
MOVE 1000 TO LIFETIM IN WAIT-NAME.
CALL "YWAIT" USING WAIT-NAME APP-NAME CONN-NAME BEF-NAME.

*****
*   EVALUATE RECEIVED EVENT                   *
*****
IF LETTER IN WAIT-NAME
THEN GO TO LETTER-PROCESSING
ELSE NEXT SENTENCE.

IF GOSIGNAL IN WAIT-NAME
THEN GO TO GOSIGNAL-PROCESSING
ELSE NEXT SENTENCE.

IF LOSCON IN WAIT-NAME
THEN GO TO LOSCON-PROCESSING.

IF NOEVENT IN WAIT-NAME
THEN GO TO WAIT-FOR-EVENT
ELSE NEXT SENTENCE.

LETTER-PROCESSING.
*****
*   RETURN RECEIVED MESSAGE.                 *
*****
IF RCD IN BEF-NAME EQUAL 0
THEN
    PERFORM SEND,
    GO TO WAIT-FOR-EVENT
ELSE
    GO TO END.

LOSCON-PROCESSING.
*****
*   CONNECTION LOST.                         *
*****
GO TO END.

```

```

GOSIGNAL-PROCESSING.
*****
*   GOSIGNAL RECEIVED.                               *
*****
      PERFORM SEND.
      GO TO WAIT-FOR-EVENT.

SEND SECTION.
S1.
*****
*   SEND MESSAGE.                                     *
*****

      CALL "YSEND" USING APP-NAME CONN-NAME BEF-NAME IOAREA.
      IF RCD IN BEF-NAME NOT EQUAL 0
      THEN
          IF RCD IN BEF-NAME NOT EQUAL 16
          OR ECD IN BEF-NAME NOT EQUAL 4
          OR IND IN BEF-NAME NOT EQUAL 12
          THEN
*****
*   IF FDBK NOT EQUAL WAIT-GO AND OK TERMINATE      *
*****
          GO TO END
          ELSE
*****
*   FIRST WAIT FOR GO-SIGNAL, THEN REPEAT SEND ATTEMPT *
*****
          NEXT SENTENCE
          ELSE
              PERFORM RECEIVE-ASY.

RECEIVE-ASY SECTION.
R1.
      MOVE "YES" TO SYN IN BEF-NAME.
      MOVE "YES" TO Q IN BEF-NAME.
      MOVE 1200 TO TOVAL IN BEF-NAME.
      MOVE 4096 TO LENGTH IN IOAREA.

      CALL "YRECEIVE" USING APP-NAME CONN-NAME BEF-NAME
          IOAREA.
      IF RCD IN BEF-NAME NOT EQUAL 0
      THEN GO TO END.

E1 SECTION.
END.
*****
*   CLEAR DOWN CONNECTION.                             *
*****

      CALL "YCLSCON" USING APP-NAME CONN-NAME.

*****
*   CLOSE APPLICATION.                                 *
*****
      CALL "YCLOSE" USING APP-NAME.
      STOP RUN.

```

5.2 DCAM(NEA) transport service

The "PYTHAGOR" application waits for a connection request from the "MEDIATOR" application with a specific connection message ("YINQUIRE TOPLOGON" or "YOPNCON ACQUIRE"). If both conditions are met, the request is accepted ("YOPNCON ACCEPT").

"PYTHAGOR" then sends an input request to "MEDIATOR" which is displayed on the monitor. Following an input, this request is returned to "PYTHAGOR" where the calculation is performed and the result reported to "MEDIATOR".

If the end condition occurs both applications are terminated.

"MEDIATOR" application

```

ID DIVISION.
PROGRAM-ID. MEDIATOR.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
    COPY YDDCUAPL.
    COPY YDDCUCON.
    COPY YDDCUCOM.
    COPY YDDCUDIS.
01  V-MESSAGE.
    02  LENGTH          PIC 9(4) COMP VALUE IS ZERO.
    02  TXT             PIC X(80) VALUE IS SPACES.
01  SEND-AREA.
    02  SEND-LENGTH    PIC 9(4) COMP VALUE IS ZERO.
    02  SEND-TEXT      PIC X(8)  VALUE IS SPACES.
01  REC-AREA.
    02  REC-LENGTH     PIC 9(4) COMP VALUE IS ZERO.
    02  REC-TEXT       PIC X(450) VALUE IS SPACES.
77  RESPONSE          PIC X(8)  VALUE IS SPACES.
77  EK                PIC X      VALUE IS SPACES.
PROCEDURE DIVISION.
OPEN.
*      "MEDIATOR" APPLICATION IS OPENED.
      MOVE "MEDIATOR" TO APPNAME IN APP-NAME.
      CALL "YOPEN" USING APP-NAME.
      IF RCD IN APP-NAME IS = 0 THEN NEXT SENTENCE
      ELSE GO TO PAR3.
      MOVE "PYTHAGOR" TO PTNNAME IN CONN-NAME.
      MOVE SPACES TO PRONAME IN CONN-NAME.
      MOVE "YES" TO SPEC IN BEF-NAME.
      MOVE "NO" TO ACCPT IN BEF-NAME.
      MOVE "YES" TO CS IN BEF-NAME.
      MOVE 80 TO LENGTH IN V-MESSAGE.
*
*      THE CONNECTION REQUEST IS TO BE ACCOMPANIED BY A
*      CONNECTION MESSAGE. THE MESSAGE IS ENTERED
*      IN THE TXT FIELD IN V-MESSAGE.
*
*
*

```



```
MOVE "MEDIATOR APPLICATION WAITING FOR ACCEPTANCE" TO TXT
    IN V-MESSAGE.
*
* CONNECTION REQUEST TO PYTHAGOR
*
CALL "YOPNCON" USING APP-NAME CONN-NAME BEF-NAME
    VTLG-NAME V-MESSAGE.
IF RCD IN BEF-NAME IS = 0 THEN NEXT SENTENCE
ELSE GO TO PAR2.
MOVE 600 TO TOVAL IN BEF-NAME.
MOVE "YES" TO Q IN BEF-NAME.
MOVE 450 TO REC-LENGTH IN REC-AREA.
*
DIALOG.
IF EK IS NOT = "E" THEN NEXT SENTENCE
ELSE GO TO END.
CALL "YRECEIVE" USING APP-NAME CONN-NAME BEF-NAME
    REC-AREA.
IF RCD IN BEF-NAME IS = 0 THEN NEXT SENTENCE
ELSE GO TO PAR1.
DISPLAY REC-TEXT UPON TERMINAL.
MOVE 8 TO SEND-LENGTH IN SEND-AREA.
MOVE SPACES TO RESPONSE SEND-TEXT IN SEND-AREA
    REC-TEXT.
*
* INPUT OF LEG LENGTHS.
*
ACCEPT RESPONSE FROM TERMINAL.
MOVE RESPONSE TO SEND-TEXT IN SEND-AREA.
CALL "YSEND" USING APP-NAME CONN-NAME BEF-NAME
    SEND-AREA.
IF RCD IN BEF-NAME IS = 0 THEN NEXT SENTENCE
ELSE GO TO PAR1.
GO TO DIALOG.
PAR1.
PERFORM FEEDBACK2.
MOVE "E" TO EK.
GO TO DIALOG.
PAR2.
PERFORM FEEDBACK2.
MOVE "E" TO EK.
GO TO END.
PAR3.
PERFORM FEEDBACK1.
MOVE "E" TO EK.
GO TO END.
FEEDBACK1.
DISPLAY "ACB-RCD: " RCD IN APP-NAME UPON TERMINAL.
DISPLAY "ACB-ECD: " ECD IN APP-NAME UPON TERMINAL.
FEEDBACK2.
DISPLAY "RPB-RCD: " RCD IN BEF-NAME UPON TERMINAL.
DISPLAY "RPB-ECD: " ECD IN BEF-NAME UPON TERMINAL.
DISPLAY "RPB-IND: " IND IN BEF-NAME UPON TERMINAL.
END.
STOP RUN.
```

"Pythagor" application

```

ID DIVISION.
PROGRAM-ID. PYTHAGOR.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
/
        COPY YDDCUAPL.
/
        COPY YDDCUCON.
/
        COPY YDDCUCOM.
/
*
*   FUNCTION VARIABLE FOR YINQUIRE
*
77  FUNC                PIC X(3) VALUE IS SPACES.
*
*   AREA FOR RESULT OF YINQUIRE TOPLOGON
*
01  TOPL-AREA.
     02  TOPL-LENGTH    PIC 9(4) COMP.
     02  PTNCHAR.
           03  PTNTYPE    PIC 9(4) COMP.
           03  DEVTYPE    PIC 9(4) COMP.
           03  CHARSET    PIC 9(4) COMP.
           03  DEVSEC     PIC 9(4) COMP.
     02  PEDIT          PIC X(3).
     02  PPROC          PIC X(3).
*
*   CONNECTION MESSAGE
*
01  LGMSG-AREA.
     02  LGMSG-LENGTH   PIC 9(4) COMP VALUE IS ZERO.
     02  LGMSG          PIC X(80) VALUE IS SPACES.
*
*   SEND AND RECEIVE AREAS
*
01  SEND-AREA.
     02  SEND-LENGTH    PIC 9(4) COMP VALUE IS ZERO.
     02  SEND-TEXT1     PIC X(450) VALUE IS SPACES.
01  SEND-TEXT2.
     02  SEND-TEXT3     PIC X(31) VALUE IS SPACES.
     02  SEND-TEXT4     PIC X(7)  VALUE IS SPACES.
     02  FILLER         PIC X(122) VALUE IS SPACES.
     02  SEND-TEXT5-1   PIC X(80) VALUE IS SPACES.
     02  SEND-TEXT5-2   PIC X(80) VALUE IS SPACES.
01  SEND-TEXT6.
     02  SEND-TEXT7     PIC X(160) VALUE IS SPACES.
     02  SEND-TEXT8     PIC X(160) VALUE IS SPACES.
     02  SEND-TEXT9     PIC X(80)  VALUE IS SPACES.
01  REC-AREA.
     02  REC-LENGTH     PIC 9(4) COMP VALUE IS ZERO.
     02  REC-TEXT1.
           03  REC-TEXT2   PIC X VALUE IS SPACES.
           03  REC-TEXT3   PIC X(4) VALUE IS SPACES.
     02  FILLER         PIC X(3).

```

```

*
* 'CALCULATION FIELDS'
*
77 H1                PIC 9(5) VALUE IS ZERO.
77 H                 PIC ZZ9.99.
01 NUMBERS.
   02 K1             PIC 99 VALUE IS ZERO.
   02 FILLER        PIC X.
   02 K2             PIC 99 VALUE IS ZERO.
*
* INPUT FOR 'PYTHAGOR' APPLICATION
*
77 RESPONSE          PIC X VALUE IS SPACE.
*
* VARIOUS END IDENTIFIERS
*
77 EK                PIC X VALUE IS SPACE.
77 EKR              PIC X VALUE IS SPACE.
77 EN                PIC X VALUE IS SPACE.
/
PROCEDURE DIVISION.
OPEN.
*****
* PYTHAGOR APPLICATION IS OPENED. *
*****
   MOVE "PYTHAGOR" TO APPNAME IN APP-NAME.
   CALL "YOPEN" USING APPNAME.
   IF RCD IN APP-NAME IS EQUAL TO 0 THEN GO TO PAR1
   ELSE PERFORM FEEDBACK1
   MOVE "E" TO EK
   GO TO INQUIRY.
PAR1.
   MOVE "TOP" TO FUNC.
   MOVE 14 TO TOPL-LENGTH IN TOPL-AREA.
INQUIRY.
*****
* THE FIRST ENTRY IN A REQUEST QUEUE IS CHECKED IN A *
* LOOP. HE REQUEST IS ACCEPTED IF THE CONNECTION *
* MESSAGE "MEDIATOR APPLICATION WAITING FOR ACCEPTANCE" *
* IS TRANSMITTED. OTHERWISE THE LOOP IS PASSED THROUGH *
* AGAIN. *
*****
   IF EK IS NOT = "E" AND EN IS NOT = "E" THEN NEXT
   SENTENCE
   ELSE GO TO DIALOG.
   MOVE 80 TO LGMSG-LENGTH IN LGMSG-AREA
   CALL "YINQUIRE" USING FUNC APP-NAME CONN-NAME
   TOPL-AREA LGMSG-AREA.
   IF RCD IN APP-NAME IS = 0 THEN NEXT SENTENCE
   ELSE PERFORM FEEDBACK1
   MOVE "E" TO EK
   GO TO INQUIRY.
   IF PTNNAME IN CONN-NAME IS NOT = "MEDIATOR"
   DISPLAY "REQUEST FROM " PTNNAME IN CONN-NAME
   "/END=E; INQUIRY=N" UPON TERMINAL
   ACCEPT RESPONSE FROM TERMINAL
   IF RESPONSE IS NOT = "E"
   THEN GO TO INQUIRY

```

```

ELSE MOVE "E" TO EK
GO TO INQUIRY
ELSE IF LGMSG IN LGMSG-AREA IS =
      "MEDIATOR APPLICATION WAITING FOR ACCEPTANCE"
MOVE 9999 TO TOVAL IN BEF-NAME
MOVE "YES" TO Q IN BEF-NAME
MOVE "YES" TO CS IN BEF-NAME
MOVE "YES" TO SPEC IN BEF-NAME
CALL "YOPNCON" USING APP-NAME CONN-NAME BEF-NAME
IF RCD IN BEF-NAME IS = 0
MOVE 450 TO SEND-LENGTH IN SEND-AREA
MOVE "E" TO EN
GO TO INQUIRY
ELSE PERFORM FEEDBACK2
MOVE "E" TO EK
GO TO INQUIRY
ELSE DISPLAY "V-MESSAGE INVALID, V-MESSAGE IS "
LGMSG UPON TERMINAL
MOVE "E" TO EK
GO TO INQUIRY.

/
DIALOG.
*****
*   THIS LOOP SERVES TO SEND AND RECEIVE MESSAGES.   *
*****
      IF EK IS NOT EQUAL TO "E" THEN NEXT SENTENCE
      ELSE GO TO END.
      MOVE SPACES TO EKR.
      MOVE "THE PROGRAM SPECIFIES THE LENGTH OF THE HYPOTENUSE OF A"
      "RIGHT-ANGLED TRIANGLE. THE LENGTH OF THE HYPOTENUSE IS COMPUTED"
      "USING THE PYTHAGOREAN THEOREM;" TO SEND-TEXT7.
      MOVE "SQUARE OF HYPOTENUSE = SUM OF THE SQUARES OF THE LEGS:"
      TO SEND-TEXT8.
      MOVE "ENTER TWO NUMBERS BETWEEN 01 AND 99 (E.G.:
      "'03 04')" TO SEND-TEXT9.
      MOVE SPACES TO SEND-TEXT1.
      MOVE SEND-TEXT6 TO SEND-TEXT1.
SEND.
      IF EK IS NOT = "E" AND EKR IS NOT = "E"
      THEN NEXT SENTENCE
      ELSE GO TO END.
      CALL "YSEND" USING APP-NAME CONN-NAME BEF-NAME
      SEND-AREA.
      IF RCD IN BEF-NAME IS = 0 THEN NEXT SENTENCE
      ELSE PERFORM FEEDBACK2
      MOVE "E" TO EK
      GO TO SEND.
      MOVE 8 TO REC-LENGTH IN REC-AREA.
RECEIVE.
      CALL "YRECEIVE" USING APP-NAME CONN-NAME BEF-NAME
      REC-AREA.
      IF RCD IN BEF-NAME IS = 0 THEN NEXT SENTENCE
      ELSE PERFORM FEEDBACK2
      MOVE "E" TO EK
      GO TO SEND.

*
*   THE FIRST CHARACTER OF THE RECEIVE TEXT IS CHECKED.
*   SUBSEQUENTLY IT IS DECIDED WHETHER AND HOW PROCESSING

```

```

*           IS TO BE CONTINUED.
*
IF REC-TEXT2 IN REC-AREA IS = "J" GO TO CHECK1.
IF REC-TEXT2 IN REC-AREA IS = "N" GO TO CHECK2.
IF REC-TEXT2 IN REC-AREA IS NOT NUMERIC GO TO CHECK3.
IF REC-TEXT2 IN REC-AREA IS NUMERIC GO TO CHECK4.
CHECK1.
  MOVE "E" TO EK
  GO TO SEND.
CHECK2.
  MOVE "E" TO EKR
  GO TO SEND.
CHECK3.
  MOVE SPACES TO SEND-TEXT1
  MOVE "INVALID INPUT" TO SEND-TEXT1
  GO TO SEND.
CHECK4.
  PERFORM PROCESSING
  GO TO SEND.
END.
  GO TO DIALOG.
PROCESSING.
*****
*           THE LENGTH OF THE HYPOTENUSE IS COMPUTED.           *
*****
  MOVE SPACES TO NUMBERS.
  MOVE REC-TEXT1 IN REC-AREA TO NUMBERS.
  COMPUTE H1 = K1 * K1 + K2 * K2.
  COMPUTE H = H1 ** (1 / 2).
  MOVE SPACES TO SEND-TEXT1 IN SEND-AREA.
  MOVE "THE LENGTH OF THE HYPOTENUSE IS" TO SEND-TEXT3.
  MOVE H TO SEND-TEXT4.
  MOVE "END? ANSWER Y (FOR YES) OR N (FOR NO)" TO
  SEND-TEXT5-1.
  MOVE "          OR ENTER TWO OTHER NUMBERS" TO
  SEND-TEXT5-2.
  MOVE SEND-TEXT2 TO SEND-TEXT1.
  MOVE SPACES TO SEND-TEXT2.
*****
*           OUTPUT ERRORS REPORTED BY DCAM.                       *
*****
  FEEDBACK1.
    DISPLAY "ACB-RCD: " RCD IN APP-NAME UPON TERMINAL.
    DISPLAY "ACB-ECD: " ECD IN APP-NAME UPON TERMINAL.
  FEEDBACK2.
    DISPLAY "RPB-RCD: " RCD IN BEF-NAME UPON TERMINAL.
    DISPLAY "RPB-ECD: " ECD IN BEF-NAME UPON TERMINAL.
    DISPLAY "RPB-IND: " IND IN BEF-NAME UPON TERMINAL.
*****
*           STOP RUN.                                             *
*****
END.
  STOP RUN.

```

The sample program "DDTTEST" demonstrates the use of the asynchronous calls YOPNCON and YRECEIVE, and the use of the new YINQUIRE functions.

First it requests the maximum number of connections that the application is to administer.

After opening the application, one asynchronous YOPNCON or YRECEIVE is issued. The program then waits for the incoming events in a loop.

Connection requests are accepted until the maximum number of connections is reached; the application then enters the STOP state. If a connection is cleared down by a partner, the application is returned to the START state.

After connection cleardown, the program fetches information about the partner (YINQUIRE with the functions "BTI"(BTERMINF), "MCS"(MONCHARS) and "POT"(PEROTERM)) and displays it on the terminal.

If the last partner has cleared down the connection ("END") or one of the partners has entered "TERM", the application is closed.

Any other inputs are simply returned to the partner (YSEND).

All DCAM calls are logged continually on the terminal, likewise the number of cleared down connections (YINQUIRE "CNT"(COUNTPTN)).

```
ID DIVISION.
PROGRAM-ID. DDTTEST.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.

        COPY YDDCUAPL REPLACING APP-NAME BY ACB.
/
        COPY YDDCUCON REPLACING CONN-NAME BY CCB.
/
        COPY YDDCUCOM REPLACING BEF-NAME BY RPB.
/
        COPY YDDCUWAI.
/
        COPY YDDCUAPL REPLACING APP-NAME BY REJ-ACB.
/
        COPY YDDCUCON REPLACING CONN-NAME BY REJ-CCB.
/
*
*   BEREICH FUER ERGEBNIS VON YINQUIRE BTERMINF
*
01  BTI-BEREICH.
    02  BTI-LAENGE      PIC 9(4) COMP.
    02  BTI-ERG.

        03  BTI-DATA.
        COPY VTSUBAS.
    03  FILLER PIC X    VALUE IS "***".
/
```

```

*
*   BEREICH FUER ERGEBNIS VON YINQUIRE MONCHARS
*
01  MCS-BEREICH.
    02  MCS-LAENGE      PIC 9(4) COMP.
    02  MCS-ERG.
        03  MCS-DATA.
        COPY VTSUMONC.
        03  FILLER PIC X   VALUE IS "*".
/
*
*   BEREICH FUER ERGEBNIS VON YINQUIRE PEROTERM
*
01  POT-BEREICH.
    02  POT-LAENGE      PIC 9(4) COMP.
    02  POT-ERG.
        03  POT-DATA.
        COPY VTSUPERP.
        03  FILLER PIC X   VALUE IS "*".
/

01  SAREA.
    02  SLAENGE          PIC 9(4) COMP VALUE IS 255.
    02  SZEILE           PIC X(255) VALUE IS SPACES.

01  INQ-MSG.
    02  FILLER           PIC X(1)  VALUE IS SPACES.
    02  AKTION           PIC X(6)  VALUE IS SPACES.
    02  INFO             PIC X(40) VALUE IS SPACES.

01  DISPMMSG.
    02  FILLER           PIC X(1)  VALUE IS SPACES.
    02  AKTION           PIC X(15) VALUE IS SPACES.
    02  PTNNAME          PIC X(8)  VALUE IS SPACES.
    02  FILLER           PIC X(1)  VALUE IS SPACES.
    02  PRONAME          PIC X(8)  VALUE IS SPACES.
    02  FILLER           PIC X(1)  VALUE IS SPACE.
    02  TEXT1            PIC X(4)  VALUE IS "LEN=".
    02  ARECLN           PIC 9(4).
    02  FILLER           PIC X(1)  VALUE IS SPACE.
    02  RZEILEDISP       PIC X(4)  VALUE IS SPACES.

01  RAREA.
    02  RLAENGE          PIC 9(4) COMP VALUE IS 255.
    02  RZEILE           PIC X(255) VALUE IS SPACES.
    02  RZEILE-REDEF1    REDEFINES RZEILE.
        03  ENDE-KRIT1    PIC X(4).
        03  FILLER        PIC X(251).
    02  RZEILE-REDEF2    REDEFINES RZEILE.
        03  ENDE-KRIT2    PIC X(1).
        03  FILLER        PIC X(254).
    02  RZEILE-REDEF3    REDEFINES RZEILE.
        03  AKTION        PIC X(8).
        03  FILLER        PIC X(247).
    02  RZEILE-REDEF4    REDEFINES RZEILE.
        03  MDATAIND      PIC X(1).
        03  FILLER        PIC X(254).
/

01  FEEDBACK-MESSAGE.

```

```
02 CC-EM2 PIC X(1) VALUE IS SPACE.
02 AKTION PIC X(15) VALUE IS SPACES.
02 FILLER PIC X(1) VALUE IS SPACE.
02 EVENT-TXT PIC X(8) VALUE IS SPACES.
02 FILLER PIC X(1) VALUE IS SPACE.
02 EVENT PIC X(8) VALUE IS SPACES.
02 CC-NL22 PIC X(1) VALUE IS SPACE.
02 FILLER PIC X(8) VALUE IS SPACES.
02 RCDTXT PIC X(6) VALUE IS " RCD: ".
02 RCD PIC X(4).
02 ECDTXT PIC X(6) VALUE IS " ECD: ".
02 ECD PIC X(4).
02 INDTXT PIC X(6) VALUE IS " IND: ".
02 IND PIC X(4).
/
01 REJLOG-MESSAGE.
02 FILLER PIC X(8) VALUE IS SPACES.
02 REJLOG-TXT PIC X(20) VALUE IS
"OPNCON REJECTED FOR ".
02 PTNNAME PIC X(8) VALUE IS SPACES.
02 FILLER PIC X(1) VALUE IS "/".
02 PRONAME PIC X(8) VALUE IS SPACES.
/
01 EREIGNIS-TEXT.
02 NOEVENT PIC X(8) VALUE IS "NOEVENT".
02 LETTER PIC X(8) VALUE IS "LETTER".
02 OPENED PIC X(8) VALUE IS "OPENED".
02 GOSIGNAL PIC X(8) VALUE IS "GOSIGNAL".
02 LOSCON PIC X(8) VALUE IS "LOSCON".
02 EVENT-TXT PIC X(8) VALUE IS " EVENT: ".
02 LOSCON-TXT PIC X(14) VALUE IS "LOSCONREASON: ".

01 REC-LN PIC 9(4) COMP VALUE IS 256.

01 DCAMVER PIC 9(4) COMP-3 VALUE IS 0800.

01 COUNTER PIC 9(4) COMP VALUE IS ZERO.

01 DUMMY-COUNTER PIC 9(4) COMP VALUE IS ZERO.

01 INIT-TABELLE PIC X(320) VALUE IS SPACES.
01 PARTNER REDEFINES INIT-TABELLE.
02 PARTNAM PIC X(8) OCCURS 20.
02 PARTPRO PIC X(8) OCCURS 20.

01 INQ-FUNKTION PIC X(3) VALUE IS "TOP".
/
01 TOP-BEREICH.
02 TOP-LAENGE PIC 9(4) COMP VALUE IS 14.
02 PTNCHAR.
03 PTNTYPE PIC 9(4) COMP VALUE IS ZERO.
03 DEVTYPE PIC 9(4) COMP VALUE IS ZERO.
03 CHARSET PIC 9(4) COMP VALUE IS ZERO.
03 DEVSEC PIC 9(4) COMP VALUE IS ZERO.
02 PEDIT PIC X(3).
02 PPROC PIC X(3).

01 LGMSG-BEREICH.
```



```

02 LGMSG-LAENGE      PIC 9(4) COMP VALUE IS 80.
02 LGMSG             PIC X(80).

01 ZAEHLER          PIC 9(4) VALUE IS 1.

01 ZAEHLER-2        PIC 9(4) VALUE IS ZERO.

01 LOSCON-ZAEHLER   PIC 9(4) VALUE IS ZERO.

01 ACTIVE-PARTNER-NO  PIC 9(4) VALUE IS ZERO.

01 OPNCON-INDICATOR PIC 9(4) VALUE IS ZERO.

01 STOP-INDIKATOR   PIC 9(4) VALUE IS ZERO.

01 BLANK-8          PIC X(8) VALUE IS SPACES.

01 MAXLN-POSITIV    PIC 9(8) COMP VALUE IS ZERO.
01 REDEFIN          REDEFINES MAXLN-POSITIV.
02 UNUSED           PIC 9(4) COMP.
02 MAXLNPOS         PIC 9(4) COMP.

01 YINQUIRE-FUNKTION PIC X(3) VALUE IS "APP".

01 APPSTAT-BEREICH.
02 LAENGE           PIC 9(4) COMP VALUE IS 4.
02 ZUSTAND          PIC 9(4) COMP VALUE IS ZERO.

01 COUNTPTN-BEREICH.
02 LAENGE           PIC 9(4) COMP VALUE IS 6.
02 VERBINDUNGSZAHL PIC 9(4) COMP VALUE IS ZERO.
02 AUFFORDERUNGSZAHL PIC 9(4) COMP VALUE IS ZERO.

01 OPNCON-FAIL PIC X(1) VALUE IS "N".
01 FELDZAEHLER PIC 9(4) VALUE IS ZERO.

01 MERKTABINIT PIC X(360) VALUE IS SPACES.
01 MERKTAB REDEFINES MERKTABINIT.
02 MERKPTN        PIC X(8) OCCURS 20.
02 MERKPRO        PIC X(8) OCCURS 20.

01 NACHRICHT-1.
02 FILLER         PIC X(10) VALUE IS "PTNNAME = ".
02 PTNNAME        PIC X(8) VALUE IS SPACE.
02 FILLER         PIC X(11) VALUE IS "/PRONAME = ".
02 PRONAME        PIC X(8) VALUE IS SPACE.
02 FILLER         PIC X(3) VALUE IS SPACE.

* _____ !ANZAHL _____
01 MAX-VERB       PIC 9      VALUE IS 2.
01 MAX-CHAR       PIC X      VALUE IS SPACE.
/
PROCEDURE DIVISION.

      MOVE SPACES TO PTNNAME IN REJ-CCB PRONAME IN REJ-CCB.

EINGABE.

```

```
MOVE "BITTE MAXIMALE ANZAHL VERBINDUNGEN EINGEBEN!"
  TO DISPMSG.
DISPLAY DISPMSG UPON TERMINAL.
MOVE "(EINE ZAHL ZWISCHEN EINS UND NEUN (1..9))"
  TO DISPMSG.
DISPLAY DISPMSG UPON TERMINAL.
ACCEPT MAX-CHAR FROM TERMINAL.
IF MAX-CHAR IS NOT NUMERIC THEN GO TO EINGABE.
MOVE MAX-CHAR TO MAX-VERB.
IF MAX-VERB LESS 1
OR MAX-VERB GREATER 9
THEN GO TO EINGABE.

/
*****
*   ANWENDUNG "DDTTEST" EROEFFNEN   *
*****

MOVE "DDTTEST" TO APPNAME IN ACB APPNAME IN REJ-ACB.
MOVE "YOPEN" TO AKTION IN FEEDBACK-MESSAGE.
CALL "YOPEN" USING ACB DCAMVER.
IF RCD IN ACB NOT EQUAL 0
THEN
  PERFORM FEEDBACK-MESSAGE-ACB
  GO TO STOP-RUN.

/
*****
*   YOPNCON-ASY ABSETZEN           *
*****
MOVE 1 TO OPNCON-INDICATOR.
PERFORM OPNCON-ASY.
*****
*   YRECEIVE ABSETZEN             *
*****
PERFORM RECEIVE-ASY.
GO TO CONTINUE-WITH-WAIT.

WAIT-FOR-EVENT.
IF ACTIVE-PARTNER-NO GREATER MAX-VERB
OR ACTIVE-PARTNER-NO EQUAL MAX-VERB
THEN
  PERFORM YINQUIRE
  IF RCD IN REJ-ACB NOT GREATER 4
  AND PTNNAME IN REJ-CCB NOT EQUAL BLANK-8
  THEN
    PERFORM YREJLOG
    ELSE NEXT SENTENCE
  ELSE NEXT SENTENCE.

MOVE SPACES TO PTNNAME IN REJ-CCB.
MOVE SPACES TO PRONAME IN REJ-CCB.

CONTINUE-WITH-WAIT.
*****
*   AUF EREIGNIS WARTEN           *
*****
MOVE 5 TO LIFETIM IN WAIT-NAME.
CALL "YWAIT" USING WAIT-NAME ACB CCB RPB.
```

```

/
*****
*   EINGETROFFENES EREIGNIS AUSWERTEN   *
*****

    IF NOEVENT IN WAIT-NAME
    THEN GO TO TRY-OPNCON-ACQ
    ELSE NEXT SENTENCE.

    IF LETTER IN WAIT-NAME
    THEN GO TO LETTER-BEHANDLUNG
    ELSE NEXT SENTENCE.

    IF OPENED IN WAIT-NAME
    THEN GO TO OPNCON-BEHANDLUNG
    ELSE NEXT SENTENCE.

    IF GOSIGNAL IN WAIT-NAME
    THEN GO TO GOSIGNAL-BEHANDLUNG
    ELSE NEXT SENTENCE.

    IF LOSCON IN WAIT-NAME
    THEN GO TO LOSCON-BEHANDLUNG
    ELSE DISPLAY "UNSINNIGES EREIGNIS" UPON TERMINAL.
    GO TO WAIT-FOR-EVENT.

/
LETTER-BEHANDLUNG.
*****
*   EINGETROFFENE NACHRICHT VERARBEITEN   *
*****
    MOVE "YRECEIVE-ASY" TO AKTION IN FEEDBACK-MESSAGE.
    PERFORM FEEDBACK-MESSAGE-RPB.

*****
*   BEI ZEITABLAUF (RCD=4,ECD=16) NEUER REC   *
*****
    IF RCD IN RPB EQUAL 4
    AND ECD IN RPB EQUAL 16
    THEN
        GO TO PERFORM-NEW-RECEIVE.

    MOVE SPACES TO DISPMSG.
    MOVE "LETTER"          TO AKTION          IN DISPMSG.
    MOVE PTNNAME IN CCB   TO PTNNAME         IN DISPMSG.
    MOVE PRONAME IN CCB   TO PRONAME         IN DISPMSG.
    MOVE ARECLN  IN RPB   TO ARECLN          IN DISPMSG.
    MOVE "LEN="          TO TEXT1            IN DISPMSG.
    MOVE ENDE-KRIT1     TO RZEILEDISP       IN DISPMSG.

    DISPLAY DISPMSG UPON TERMINAL.

    PERFORM SENDEN.

    IF ENDE-KRIT1 EQUAL "TERM"
    THEN
        MOVE "PROGRAMM DDTTEST WIRD BEEENDET ! " TO RZEILE
        MOVE 32   TO ARECLN IN RPB
        PERFORM SENDEN

```

```
GO TO ENDE.

IF ENDE-KRIT1 EQUAL "ENDE"
THEN
  MOVE "VERBINDUNG WIRD ABGEBAUT !" TO RZEILE
  MOVE 26 TO ARECLN IN RPB
  PERFORM SENDEN
  MOVE "YCLSCON" TO AKTION IN FEEDBACK-MESSAGE
  CALL "YCLSCON" USING ACB CCB
  PERFORM FEEDBACK-MESSAGE-ACB
  ADD 1 TO LOSCON-ZAEHLER
  SUBTRACT 1 FROM ACTIVE-PARTNER-NO
  IF ACTIVE-PARTNER-NO EQUAL 0
  THEN GO TO ENDE
  ELSE
    PERFORM YSETLOG
    PERFORM SUCHE-PARTNER-NAME
    PERFORM YINQUIRE-COUNTPTN
    IF OPNCON-INDICATOR EQUAL 0
    THEN
      MOVE 1 TO OPNCON-INDICATOR
      PERFORM OPNCON-ASY
    ELSE NEXT SENTENCE
  ELSE NEXT SENTENCE.

MOVE SPACES TO RZEILE.
MOVE SPACES TO SZEILE.

PERFORM-NEW-RECEIVE.

PERFORM RECEIVE-ASY.

GO TO TRY-OPNCON-ACQ.

/
TRY-OPNCON-ACQ.
  IF OPNCON-FAIL = "N"
  THEN GO TO CONTINUE-WITH-WAIT.
TRY-OPNCON-CONT.
  IF FELDZAEHLER GREATER MAX-VERB
  OR FELDZAEHLER = MAX-VERB
  THEN
    GO TO ENDE-TRY-OPNCON.

  IF MERKPTN(FELDZAEHLER) EQUAL BLANK-8
  THEN ADD 1 TO FELDZAEHLER
  ELSE
    MOVE MERKPTN(FELDZAEHLER) TO PTNNAME IN CCB
    MOVE MERKPRO(FELDZAEHLER) TO PRONAME IN CCB
    PERFORM OPNCON-ACQ
    IF RCD IN RPB EQUAL 0
    THEN
      MOVE PTNNAME IN CCB TO PTNNAME IN NACHRICHT-1
      MOVE PRONAME IN CCB TO PRONAME IN NACHRICHT-1
      MOVE NACHRICHT-1 TO RZEILE
      MOVE 40 TO ARECLN IN RPB
```

```

PERFORM SENDEN

MOVE "N" TO OPNCON-FAIL
MOVE SPACES TO MERKPTN(FELDZAEHLER)
MOVE SPACES TO MERKPRO(FELDZAEHLER)
ADD 1 TO FELDZAEHLER
PERFORM YINQUIRE-COUNTPTN
GO TO TRY-OPNCON-CONT
ELSE
MOVE "Y" TO OPNCON-FAIL
ADD 1 TO FELDZAEHLER
GO TO TRY-OPNCON-CONT.
ENDE-TRY-OPNCON.
MOVE 1 TO FELDZAEHLER.
SUCHE-MERK.
IF MERKPTN(FELDZAEHLER) EQUAL BLANK-8
THEN ADD 1 TO FELDZAEHLER
ELSE
MOVE "Y" TO OPNCON-FAIL
GO TO CONTINUE-WITH-WAIT.

IF FELDZAEHLER GREATER MAX-VERB
OR FELDZAEHLER EQUAL MAX-VERB
THEN
MOVE 1 TO FELDZAEHLER
GO TO CONTINUE-WITH-WAIT
ELSE
ADD 1 TO FELDZAEHLER
GO TO SUCHE-MERK.

/
OPNCON-BEHANDLUNG.
*****
*   OPNCON-BEARBEITUNG   *
*****
ADD 1 TO ACTIVE-PARTNER-NO.
MOVE ZERO TO OPNCON-INDICATOR.
PERFORM SUCHE-FREIES-FELD
MOVE PTNNAME IN CCB TO PARTNAM IN PARTNER(ZAEHLER)
MOVE PRONAME IN CCB TO PARTPRO IN PARTNER(ZAEHLER)
IF ACTIVE-PARTNER-NO NOT GREATER MAX-VERB
THEN
MOVE SPACES TO DISPMSG
MOVE OPENED IN EREIGNIS-TEXT TO AKTION
      IN DISPMSG
MOVE PTNNAME IN CCB TO PTNNAME IN DISPMSG
MOVE PRONAME IN CCB TO PRONAME IN DISPMSG

DISPLAY DISPMSG UPON TERMINAL

MOVE PTNNAME IN CCB TO PTNNAME IN NACHRICHT-1
MOVE PRONAME IN CCB TO PRONAME IN NACHRICHT-1
MOVE NACHRICHT-1 TO RZEILE
MOVE 40          TO ARECLN  IN RPB

PERFORM SENDEN

MOVE "BELIEBIGE EINGABE <DUE> ODER ENDE <DUE> "
      TO RZEILE

```

```

MOVE 40                TO ARECLN  IN RPB

PERFORM SENDEN

MOVE SPACES TO RZEILE

*****
*      3 NEUE INQUIRE-FUNKTIONEN (AB V10)      *
*****
PERFORM YINQUIRE-BTERMINF
PERFORM YINQUIRE-MONCHARS
PERFORM YINQUIRE-PEROTERM

IF ACTIVE-PARTNER-NO LESS MAX-VERB
THEN
    MOVE 1 TO OPNCON-INDICATOR
    PERFORM OPNCON-ASY
ELSE
    MOVE "YES" TO STP IN RPB
    MOVE "YSETLOG-STP" TO AKTION IN FEEDBACK-MESSAGE
    CALL "YSETLOG" USING ACB RPB
    PERFORM FEEDBACK-MESSAGE-RPB
    MOVE 1 TO STOP-INDIKATOR.

PERFORM YINQUIRE-COUNTPTN.

GO TO WAIT-FOR-EVENT.

/
LOSCON-BEHANDLUNG.

MOVE SPACES TO DISPMSG.
MOVE "LOSCON" TO AKTION IN DISPMSG.
MOVE PTNNAME IN CCB TO PTNNAME IN DISPMSG.
MOVE PRONAME IN CCB TO PRONAME IN DISPMSG.
DISPLAY DISPMSG UPON TERMINAL.

ADD 1 TO LOSCON-ZAEHLER.

IF ACTIVE-PARTNER-NO EQUAL 0
THEN GO TO ENDE
ELSE
    PERFORM OPNCON-ACQ.
    IF RCD IN RPB NOT EQUAL 0
    THEN
        PERFORM SUCHBLANK8
        MOVE PTNNAME IN CCB TO MERKPTN(FELDZAEHLER)
        MOVE PRONAME IN CCB TO MERKPRO(FELDZAEHLER)
        MOVE "Y" TO OPNCON-FAIL
        GO TO CONTINUE-WITH-WAIT.
    IF ACTIVE-PARTNER-NO LESS MAX-VERB
    AND OPNCON-INDICATOR EQUAL 0
    THEN
        MOVE 1 TO OPNCON-INDICATOR
        PERFORM OPNCON-ASY.
    GO TO WAIT-FOR-EVENT.

/
GOSIGNAL-BEHANDLUNG.

```

```
DISPLAY " GOSIGNAL EINGETROFFEN" UPON TERMINAL.  
GO TO WAIT-FOR-EVENT.
```

```
/
```

```
SENDEN SECTION.
```

```
S1.
```

```
MOVE RZEILE TO SZEILE.
```

```
IF ARECLN IN RPB GREATER 7
```

```
THEN
```

```
MOVE ARECLN IN RPB TO SLAENGE IN SAREA
```

```
ELSE
```

```
MOVE 8 TO SLAENGE IN SAREA.
```

```
MOVE "YSEND" TO AKTION IN FEEDBACK-MESSAGE.
```

```
CALL "YSEND" USING ACB CCB RPB SAREA.
```

```
PERFORM FEEDBACK-MESSAGE-RPB.
```

```
*****  
* WARTEN AUF GOSIGNAL *  
*****
```

```
IF RCD IN RPB NOT EQUAL 16 AND
```

```
ECD IN RPB NOT EQUAL 4
```

```
THEN
```

```
PERFORM COUNT1000
```

```
GO TO ENDE-SENDEN.
```

```
ENDE-SENDEN.
```

```
/
```

```
OPNCON-ACQ SECTION.
```

```
O1.
```

```
MOVE 0 TO MAXLN IN CCB.
```

```
MOVE "NO " TO ACCPT IN RPB.
```

```
MOVE "SYS" TO EDIT IN CCB.
```

```
MOVE "YES" TO SYN IN RPB.
```

```
MOVE "YES" TO Q IN RPB.
```

```
MOVE "YES" TO SIGNAL IN CCB.
```

```
MOVE "N" TO MDATA IN CCB.
```

```
MOVE 900 TO TOVAL IN RPB.
```

```
MOVE "YOPNCON-ACQ" TO AKTION IN FEEDBACK-MESSAGE.
```

```
CALL "YOPNCON" USING ACB CCB RPB.
```

```
PERFORM FEEDBACK-MESSAGE-RPB.
```

```
/
```

```
OPNCON-ASY SECTION.
```

```
O1.
```

```
MOVE 0 TO MAXLN IN CCB.
```

```
MOVE "YES" TO ACCPT IN RPB.
```

```
MOVE "SYS" TO EDIT IN CCB.
```

```
MOVE "NO" TO SYN IN RPB.
```

```
MOVE "YES" TO Q IN RPB.
```

```
MOVE "YES" TO SIGNAL IN CCB.
```

```
MOVE 900 TO TOVAL IN RPB.
```

```
MOVE "YOPNCON-ASY" TO AKTION IN FEEDBACK-MESSAGE.
```

```
CALL "YOPNCON" USING ACB CCB RPB.
```

```
PERFORM FEEDBACK-MESSAGE-RPB.
```

```
/
```

```
RECEIVE-ASY SECTION.
```

```
R1.
  MOVE "NO" TO SYN IN RPB.
  MOVE "YES" TO Q IN RPB.
  MOVE 900 TO TOVAL IN RPB.
  MOVE REC-LN TO RLAENGE IN RAREA.
  MOVE "YRECEIVE-ASY" TO AKTION IN FEEDBACK-MESSAGE.
  CALL "YRECEIVE" USING ACB CCB RPB RAREA.
  PERFORM FEEDBACK-MESSAGE-RPB.
/
YINQUIRE SECTION.
YINQ.

  MOVE "YINQUIRE-TOP" TO AKTION IN FEEDBACK-MESSAGE.
  CALL "YINQUIRE" USING INQ-FUNKTION REJ-ACB REJ-CCB
    TOP-BEREICH.
  PERFORM FEEDBACK-MESSAGE-REJ-ACB.
/
YSETLOG SECTION.
YSLG.

  IF STOP-INDIKATOR EQUAL 1
  THEN
    MOVE "NO" TO STP IN RPB
    MOVE "YSETLOG-STA" TO AKTION IN FEEDBACK-MESSAGE
    CALL "YSETLOG" USING ACB RPB
    PERFORM FEEDBACK-MESSAGE-RPB
    MOVE 0 TO STOP-INDIKATOR
  ELSE NEXT SENTENCE.
/
YREJLOG SECTION.
YREJ.

  MOVE "YREJLOG" TO AKTION IN FEEDBACK-MESSAGE.
  CALL "YREJLOG" USING REJ-ACB REJ-CCB.
  PERFORM FEEDBACK-MESSAGE-REJ-ACB.

  MOVE PTNNAME IN REJ-CCB TO PTNNAME IN REJLOG-MESSAGE.
  MOVE PRONAME IN REJ-CCB TO PRONAME IN REJLOG-MESSAGE.

  DISPLAY REJLOG-MESSAGE UPON TERMINAL.
/
SUCHE-PARTNER-NAME SECTION.
S1.
  MOVE 1 TO ZAEHLER.
SUCH-LOOP.
  IF PARTNAM IN PARTNER(ZAEHLER) EQUAL PTNNAME IN CCB
  THEN
    MOVE SPACES TO PARTNAM IN PARTNER(ZAEHLER)
    MOVE SPACES TO PARTPRO IN PARTNER(ZAEHLER)
    GO TO ENDE-SUCHE-PARTNER-NAME
  ELSE
    ADD 1 TO ZAEHLER
    IF ZAEHLER GREATER MAX-VERB
    THEN DISPLAY " P R O G R A M M F E H L E R " UPON TERMINAL
    GO TO ENDE
  ELSE
    GO TO SUCH-LOOP.
```



```

ENDE-SUCHE-PARTNER-NAME.
/
SUCHE-FREIES-FELD SECTION.
SFF.
  MOVE 1 TO ZAEHLER.
SUCHE-WEITER.
  IF PARTNAM IN PARTNER(ZAEHLER) EQUAL BLANK-8
  THEN
    GO TO ENDE-SUCHE-FREIES-FELD
  ELSE
    ADD 1 TO ZAEHLER
    IF ZAEHLER GREATER MAX-VERB
    THEN
      GO TO ENDE
    ELSE
      GO TO SUCHE-WEITER.
ENDE-SUCHE-FREIES-FELD.

/
FEEDBACK-MESSAGE-ACB SECTION.
FMA.
  MOVE RCD IN ACB TO RCD IN FEEDBACK-MESSAGE.
  MOVE ECD IN ACB TO ECD IN FEEDBACK-MESSAGE.
  MOVE IND IN ACB TO IND IN FEEDBACK-MESSAGE.
  MOVE SPACES TO EVENT-TXT IN FEEDBACK-MESSAGE.
  MOVE SPACES TO EVENT IN FEEDBACK-MESSAGE.
  DISPLAY FEEDBACK-MESSAGE UPON TERMINAL.
FEEDBACK-MESSAGE-REJ-ACB SECTION.
FMRA.
  MOVE RCD IN REJ-ACB TO RCD IN FEEDBACK-MESSAGE.
  MOVE ECD IN REJ-ACB TO ECD IN FEEDBACK-MESSAGE.
  MOVE IND IN REJ-ACB TO IND IN FEEDBACK-MESSAGE.
  MOVE SPACES TO EVENT-TXT IN FEEDBACK-MESSAGE.
  MOVE SPACES TO EVENT IN FEEDBACK-MESSAGE.
  DISPLAY FEEDBACK-MESSAGE UPON TERMINAL.

/
YINQUIRE-COUNTPTN SECTION.
YICTP.

*****
*   YINQUIRE COUNTPTN   *
*****
  MOVE "CNT" TO YINQUIRE-FUNKTION.
  CALL "YINQUIRE" USING YINQUIRE-FUNKTION ACB CCB
    COUNTPTN-BEREICH.
  IF RCD IN ACB EQUAL 0
  THEN
*   MOVE "YINQUIRE" TO AKTION IN DISPMSG
*   MOVE "RCD: "    TO RCDT IN DISPMSG
*   MOVE "ECD: "    TO RCDT IN DISPMSG
*   MOVE "IND: "    TO RCDT IN DISPMSG
*   MOVE RCD IN ACB TO RCD  IN DISPMSG
*   MOVE ECD IN ACB TO ECD  IN DISPMSG
*   MOVE IND IN ACB TO IND  IN DISPMSG
*   DISPLAY DISPMSG UPON TERMINAL
*   ELSE

  DISPLAY " ANZAHL DER VERBINDUNGEN = " VERBINDUNGSZAHL IN

```

COUNTPTN-BEREICH UPON TERMINAL.

```
IF AUFFORDERUNGSZAHL NOT EQUAL 0
THEN
  DISPLAY " AUFFORDERUNGSZAHL = " AUFFORDERUNGSZAHL IN
  COUNTPTN-BEREICH UPON TERMINAL.
```

YINQUIRE-BTERMINF SECTION.
YIBTI.

```
*****
*   YINQUIRE BTERMINF                               *
*****
  MOVE SPACES TO BTI-DATA.
  MOVE "BTI" TO YINQUIRE-FUNKTION.
  MOVE 24 TO BTI-LAENGE IN BTI-BEREICH.
  CALL "YINQUIRE" USING YINQUIRE-FUNKTION ACB CCB
  BTI-BEREICH.

  IF RCD IN ACB EQUAL 0
  THEN
    MOVE "BTI : " TO AKTION IN INQ-MSG
    MOVE BTI-ERG TO INFO IN INQ-MSG
    DISPLAY INQ-MSG UPON TERMINAL
  ELSE
    MOVE "YINQUIRE-BTI" TO AKTION IN FEEDBACK-MESSAGE
    PERFORM FEEDBACK-MESSAGE-ACB.
```

YINQUIRE-MONCHARS SECTION.
YIMCS.

```
*****
*   YINQUIRE MONCHARS                               *
*****
  MOVE SPACES TO MCS-DATA.
  MOVE "MCS" TO YINQUIRE-FUNKTION.
  MOVE 14 TO MCS-LAENGE IN MCS-BEREICH.
  CALL "YINQUIRE" USING YINQUIRE-FUNKTION ACB CCB
  MCS-BEREICH.

  IF RCD IN ACB EQUAL 0
  THEN
    MOVE "MCS : " TO AKTION IN INQ-MSG
    MOVE MCS-ERG TO INFO IN INQ-MSG
    DISPLAY INQ-MSG UPON TERMINAL
  ELSE
    MOVE "YINQUIRE-MCS" TO AKTION IN FEEDBACK-MESSAGE
    PERFORM FEEDBACK-MESSAGE-ACB.
```

YINQUIRE-PEROTERM SECTION.
YIPOT.

```
*****
*   YINQUIRE PEROTERM                               *
*****
  MOVE SPACES TO POT-DATA.
  MOVE "POT" TO YINQUIRE-FUNKTION.
  MOVE 8 TO POT-LAENGE IN POT-BEREICH.
  CALL "YINQUIRE" USING YINQUIRE-FUNKTION ACB CCB
  POT-BEREICH.
```

```
IF RCD IN ACB EQUAL 0
THEN
  MOVE "POT : " TO AKTION IN INQ-MSG
  MOVE POT-ERG TO INFO IN INQ-MSG
  DISPLAY INQ-MSG UPON TERMINAL
ELSE
  MOVE "YINQUIRE-POT" TO AKTION IN FEEDBACK-MESSAGE
  PERFORM FEEDBACK-MESSAGE-ACB.

/
SUCHBLANK8 SECTION.
SB8.
  MOVE 1 TO FELDZAEHLER.
SUCHBLANK8ANF.
  IF MERKPTN(FELDZAEHLER) EQUAL BLANK-8
  THEN
    GO TO ENDE-SUCHBLANK8
  ELSE ADD 1 TO FELDZAEHLER
    IF FELDZAEHLER GREATER MAX-VERB
    OR FELDZAEHLER EQUAL MAX-VERB
    THEN
      MOVE 1 TO FELDZAEHLER
      GO TO ENDE-SUCHBLANK8
    ELSE
      ADD 1 TO FELDZAEHLER
      GO TO SUCHBLANK8ANF.
ENDE-SUCHBLANK8.

/
COUNT1000 SECTION.
CNT.
CNT1000.
  ADD 1 TO DUMMY-COUNTER.
  IF DUMMY-COUNTER NOT GREATER 5000
  THEN
    GO TO CNT1000
  ELSE
    MOVE ZERO TO DUMMY-COUNTER.

/
FEEDBACK-MESSAGE-RPB SECTION.
FR1.
  MOVE RCD IN RPB TO RCD IN FEEDBACK-MESSAGE.
  MOVE ECD IN RPB TO ECD IN FEEDBACK-MESSAGE.
  MOVE IND IN RPB TO IND IN FEEDBACK-MESSAGE.
  MOVE SPACES TO EVENT-TXT IN FEEDBACK-MESSAGE.
  MOVE SPACES TO EVENT IN FEEDBACK-MESSAGE.
  DISPLAY FEEDBACK-MESSAGE UPON TERMINAL.

/
FEEDBACK-MESSAGE-WAIT SECTION.
FW1.
  MOVE RSI IN WAIT-NAME TO RCD IN FEEDBACK-MESSAGE.
  MOVE RRS IN WAIT-NAME TO ECD IN FEEDBACK-MESSAGE.
  MOVE EVENT-TXT IN EREIGNIS-TEXT TO EVENT-TXT IN
  FEEDBACK-MESSAGE.
  IF NOEVENT IN WAIT-NAME
  THEN MOVE NOEVENT IN EREIGNIS-TEXT TO EVENT IN
  FEEDBACK-MESSAGE
  ELSE
    IF LETTER IN WAIT-NAME
```

```
THEN MOVE LETTER IN EREIGNIS-TEXT TO EVENT IN
FEEDBACK-MESSAGE
ELSE
  IF OPENED IN WAIT-NAME
  THEN MOVE OPENED IN EREIGNIS-TEXT TO EVENT IN
  FEEDBACK-MESSAGE
  ELSE
    IF GOSIGNAL IN WAIT-NAME
    THEN MOVE GOSIGNAL IN EREIGNIS-TEXT TO EVENT IN
    FEEDBACK-MESSAGE
    ELSE
      IF LOSCON IN WAIT-NAME
      THEN MOVE LOSCON IN EREIGNIS-TEXT TO EVENT IN
      FEEDBACK-MESSAGE
      ELSE MOVE "UNSINN" TO EVENT IN FEEDBACK-MESSAGE.

DISPLAY FEEDBACK-MESSAGE UPON TERMINAL.

ENDE SECTION.
E1.
  MOVE "YCLOSE" TO AKTION IN FEEDBACK-MESSAGE.
  CALL "YCLOSE" USING ACB.
  PERFORM FEEDBACK-MESSAGE-ACB.
  GO TO STOP-RUN.
STOP-RUN SECTION.
SR.
  STOP RUN.
```

6 Appendix

6.1 Format of the data structures

Format of the application structure						Used by 3)													
Level numbers, record names and data names	PICTURE clause	Parameter 1)	ISO	NEA	Meaning	Y O P E N	Y C L O N S E O N	Y O P L N S C O N G	Y C L E J L L O G	Y R E T L N O G E	Y S E H A Q U I R E	Y I N Q U I R E	Y P R O C E S S I D	Y S E R V E R I T Y	Y R E C O R D I N G	Y R E S O U R C E S	Y R E F E R E N C E	Y W R I T I N G	
01 APP-NAME			x	x	Record name														
02 APPNAME	PIC X(8).	application -name 2)	x	x	Name of DCAM application	x	x	x	x	x	x	x	x	x	x	x	x	x	x
02 PRONAME	PIC X(8).	processor- name	x	x	Name of home processor	+													
02 DISNAME	PIC X(8).	distribu- tion-name	-	x	Name of distribution code assigned	x													
02 LINK	PIC X(8).	link-name	x	x	Link name														
02 LINKMOD	PIC X(3).	{ "PER" "TEM" }	x	x	Permanent or temporary acceptance of CLT entries in application structure	x													
02 LOGPASS	PIC X(4).	password 1	-	x	Connection password (predefined), for DCAM(ISO):LOW-VALUE	x													
02 USEPASS	PIC X(4).	password1/2	x	x	Application password (predefined)	x													
02 USEPW	PIC X(4).	password2/3	x	x	Application password (specified)	x													
02 VERIFY	PIC X(3).	{ "PRM" "SEC" "NO" }	x	x	Primary task (=PRIMARY) Secondary task (=SECONDARY) No check (=NO)	x													
02 ATTR.			x	x	Attributes of the application														
03 SHARE	PIC X(3).	{ "YES" "NO" }	x	x	Shareable (=SHARE) Non-shareable (=NSHARE)	x													

Format of the application structure						Used by 3)																
Level numbers, record names and data names	PICTURE clause	Parameter 1)	ISO	NEA	Meaning	Y O P E N	Y C L O N S E O N	Y O P L O N S C O N G	Y C L O N S C O N G	Y R E J E C T O N G	Y S E L E C T O N G	Y C H I Q U I R E	Y I N Q U I R E	Y P R I M I T A C K	Y F E E D B A C K	Y R E J E C T O N G	Y R E J E C T O N G	Y R E J E C T O N G	Y R E J E C T O N G	Y R E J E C T O N G	Y R E J E C T O N G	
03 LOGON	PIC X(3).	{ "YES" } { "NO" }	-	x	Accept (=LOGON) } Reject (=NLOGON) }	x																
03 DISCO	PIC X(3).	{ "YES" } { "NO" }	-	x	Distribution codes used (=DISCO) No distribution codes used (=NDISCO)	x																
03 TACK	PIC X(3).	{ "PRI" } { "REQ" } { "NO" }	x	x	Ack. for primary task (=PRIMTASK) Ack.for requesting task (=REQTASK) No acknowledgments (=NOTACK)	x																
02 FDBK SYNCHRONIZED.			x	x	Feedback information																	
03 RCD	PIC 9(4) COMP.		x	x	Return code	+	+	+	+	+	+	+	+									
03 ECD	PIC 9(4) COMP.		x	x	Error code	+	+	+	+	+	+	+	+									
03 IND	PIC 9(4) COMP.		x	x	Indicator																	
02 ISO	PIC X.	"Y" 2)	x	-	DCAM(ISO) transporton	x																
02 FILLER	PIC X(8).		x	x																		

1) Aside from the exception noted, all entries are optional, some only conditionally so, however.

2) Mandatory entry

- 3) x Entered by user
- + Entered by DCAM (feedback information)
- * Entered either by user or by DCAM (feedback information), depending on application.

Format of the instruction structure						Used by 2)													
Level numbers, record names and data names	PICTURE clause	Parameter 1)	ISO	NEA	Meaning	Y O P E N	Y C L O N S E	Y O N C O N	Y C L O N G	Y R E J L O G	Y S E T A N O G E	Y I N Q U I R E	Y P E R M I T I D	Y F C R N D	Y S E N C E I V E	Y R E C E I T	Y R E C E I T	Y W A I T	
01 BEF-NAME.			x	x	Record name														
02 TOVAL	PIC 9(4) COMP.	{(0<n <9999) 600}	x	x	Maximum wait time				x								x		
02 SEQNO	PIC 9(4) COMP.	{0<number ≤9999}	-	x	Sequence number of output message										x				
02 OPTCD.			x	x	Specific. for instruct. execution														
03 SPEC	PIC X(3).	{"YES" "NO"}	x x	x x	Specific partner (=SPEC) Any partner (=ANY)			x									x	x	
03 CS	PIC X(3).	{"YES" "NO"}	x x	x x	Originator-oriented queue (=CS) Common receiver queue (=CA)			x							x	x	x		
03 Q	PIC X(3).	{"YES" "NO"}	x x	x x	Call to be queued (=Q) Call not to be queued (=NQ)			x								x			
03 ACCPT	PIC X(3).	{"YES" "NO"}	x x	x x	Accept connection request (=ACCEPT) Request connection (=ACQUIRE)				x										
03 STP	PIC X(3).	{"YES" "NO"}	- -	x x	STOP state (=STOP) START state (=START)					x									
03 DG	PIC X(3).	{"ELE" "SUB" "GRP"}	x - x	x x x	Message element (=ELEMENT) Message subgroup (=SUBGROUP) Message group (=GROUP)											x			
03 NORMAL	PIC X(3).	{"YES" "NO"}	- -	x x	Normal priority (=NORMAL) High priority (=EXPRESS)										x				
03 TRUNC	PIC X(3).	{"TRC" "KEE" "VTK"}	x x x	x x x	Cut off remainder (=TRUNC) Keep remainder (=KEEP) As specified in CONN-NAME (=CCBTK)												x		
03 TACK	PIC X(3).	{"YES" "NO"}	- -	x x	Transport acknowledgment (=TACK) No transport ack. (=NTACK)											x			
03 BELL	PIC X(3).	{"YES" "NO"}	- -	x x	Audible alarm (=BELL) No audible alarm (=NBELL)											x			
03 FHS	PIC X(3).	{"YES" "NO"}	- -	x x	Formatted I/O with FHS Formatted I/O without FHS											x	x		

Format of the instruction structure (continued)						Used by 2)														
Level numbers, record names and data names	PICTURE clause	Parameter 1)	ISO	NEA	Meaning	Y	O	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
						P	C	L	P	R	S	C	I	P	F	S	R	R	E	W
						E	L	O	O	E	E	H	N	E	C	N	E	E	E	A
						N	S	N	N	J	T	A	Q	R	R	D	C	E	I	
						S	E	C	C	L	L	N	U	M	I	I	D	E	T	
						E	O	O	O	O	O	G	E	I	R	E				
02 FDB SYNCHRONIZED.			x	x	Feedback information															
03 ASEQNO	PIC 9(4) COMP.		-	x	Sequence number of message received														+	
03 TACKNO	PIC 9(4) COMP.		-	x	Sequence number of message acknowledged														+	+
03 ARECLN	PIC 9(4) COMP.		x	x	Length of message received or of remainder being kept														+	+
03 FDBK.			x	x	Feedback information															
04 RCD	PIC 9(4) COMP.		x	x	Return code			+		+						+	+	+	+	
04 ECD	PIC 9(4) COMP.		x	x	Error code			+		+						+	+	+	+	
04 IND	PIC 9(4) COMP.		x	x	Indicator													+	+	+
03 REASON REDEFINED FDBK.			x	x																
04 LOSCON-REASON	PIC 9(4) COMP.		x	x	Feedback information on connection clear-down															+
04 FILLER	PIC X(4).		x	x																
02 OPTCD1.			x	x	Information on execution of instruction															
03 SYN	PIC X(3). COMP. {"YES"} {"NO" }		x	x	Synchronous processing (=SYN) Asynchronous processing (=ASY)				x										x	
03 VTSUCB	PIC X(3). {"YES"} {"NO" }		-	x	VTSUCB is passed													x	x	
			-	x	No VTSUCB available															
03 FILLER	PIC X(12).		x	x																
02 FILLER	PIC X(9).		x	x																

1) All user entries are optional, some only conditionally so, however.

2) x Entered by user
+ Entered by DCAM (feedback information)

Format of the connection structure					Used by 2)															
Level numbers, record names and data names	PICTURE clause	Parameter 1)	ISO	NEA	Meaning	Y O P E N	Y C L O N S E O N	Y O P L O N S E O N	Y C L O N S E O N	Y R E J L O G O G	Y S E L O G O G	Y I N Q U I R E	Y P R O C E D U R E	Y F O R M I D	Y S E N D I V E	Y R E C E I V E	Y E C S E T	Y R E C E I V E	Y W A I T	
01 CONN-NAME.			x	x	Record name															
02 PTNNAME	PIC X(8).	partner- name	x	x	Name of the partner			x	x	x	x	*			x	*	x	*	x	+
02 PRONAME	PIC X(8).	procname	x	x	Partner's processor name			x	x	x	x	*			x	*	x	*	x	+
02 PASSWORD	PIC X(4).	password 4	-	x	Connection password (specified)			x												
02 LINK	PIC X(8).	link-name	x	x	Link name			*												
02 LINKMOD	PIC X(3).	{ "PER" "TEM" }	x x	x x	Permanent or temporary acceptance of CLT entries in the connection structure			x												
02 EDITIN.			-	x	Input message editing															
03 TRANSF	PIC X(3).	{ "PHY" "LIN" "FOR" }	- - -	x x x	No virtual terminals (=PHYS) Line terminal (=LINE) Format terminal (=FORM)			x			x									
03 GETBS	PIC X(3).	{ "YES" "NO" }	- -	x x	Backspace (=GETBS) No backspace (=NGETBS)			x			x									
03 GETFC	PIC X(3).	{ "YES" "NO" }	- -	x x	The logical function key code of a terminal is the first character of the message (=GETFC); is not transmitted (=NGETFC)			x			x									
03 LCASE	PIC X(3).	{ "YES" "NO" }	- -	x x	Lower case (=LCASE) Upper case only (=NLCASE)			x			x									
02 EDITOUT.			-	x	Message editing (output)															
03 TRANSF	PIC X(3).	{ "PHY" "LIN" "FOR" }	- - -	x x x	No virtual terminals (=PHYS) Line terminal (=LINE) Format terminal (=FORM)			x			x									
03 HCOPY	PIC X(3).	{ "YES" "NO" }	- -	x x	With printout (=HCOPY) Without printout (=NHCOPY)			x			x									
03 HOM	PIC X(3).	{ "YES" "NO" }	- -	x x	All logical lines of a message are handled as a unit (=HOM) Each individual logical line is handled as a unit (=NHOM)			x			x									
02 EDIT	PIC X(3).	{ "USR" "SYS" }	- -	x x	User responsibility (=USER) System responsibility (=SYSTEM)			*												+

Format of the connection structure (continued)					Used by 2)																
Level numbers, record names and data names	PICTURE clause	Parameter 1)	ISO	NEA	Meaning	Y O P E N	Y C L O S E	Y L O N G	Y O N S C O N	Y P L S C O N	Y R E J E C T I O N	Y S E L L O G E	Y C H A R A C T E R S E T	Y I N Q U I R E	Y P R O C E D U R E	Y F O R W A R D I N G	Y S E N S I T I V E	Y R E S E R V E R	Y R E S E R V E R	Y W A I T I N G	
02 PROC.			x	x	Message handling																
03 TRUNC	PIC X(3).	{ "YES" } { "NO" }	x x	x x	Discard remainder (=TRUNC) Keep remainder (=KEEP)			x				x									
03 SYSCODE	PIC X(3).	{ "YES" } { "NO" }	- -	x x	Convert to EBCDIC (=SYSCODE) No conversion (=BINARY)				x				x								
03 APPSTART	PIC X(3).	{ "YES" } { "NO" }	- -	x x	Started by application (=APPSTART) Started by anyone (=ANYSSTART)				*												+
02 MAXLN	PIC 9(4) COMP.		x	x	Max. message length			*													+
02 PTNCHAR.			-	x	Partner characteristics - part 1																
03 PTNTYPE	PIC 9(4) COMP.		-	x	Type of partner				+												+
03 DEVTYPE	PIC 9(4) COMP.		-	x	Device type (if terminal)				+												+
03 CHARSET	PIC 9(4) COMP.		-	x	Character set of terminal				+												+
03 DEVSEC	PIC 9(4) COMP.		-	x	Optional features of terminal				+												+
02 EDITOUT1.			-	x	Message editing																
03 EXTEND	PIC X(3).	{ "YES" } { "NO" }	- -	x x	Output data protected (=EXTEND); not protected (=NEXTEND)				x				x								
03 LOGC	PIC X(3).	{ "YES" } { "NO" }	- -	x x	Logical control characters are evaluated (=LOGC); not evaluated (=NLOGC)				x				x								
03 LACK	PIC X(3).	{ "YES" } { "NO" }	- -	x x	Logical acknowledgment is requested (=LACK); not requested (=NLACK)				x				x								
03 FILLER	PIC X(15).		x	x																	

Format of the connection structure (continued)					Used by 2)														
Level numbers, record names and data names	PICTURE clause	Parameter 1)	ISO	NEA	Meaning	Y O P E N	Y C L O N S E O N	Y O P L O N S C O N G	Y C L E S T L O G	Y R E J T L O G	Y S E T A N G E	Y I N Q U I R E	Y P R O C E D U R E	Y F O R M I D	Y S E N D I V E	Y R E C E I T	Y R E C E I T	Y W A I T	
02 PROCL.			x	x	Message handling														
03 SIGNAL	PIC X(3).	{ "YES" "NO" }	x x	x x	GO signal given (=SIGNAL); not given (=NSIGNAL)			x			x								
03 TERMSTAT	PIC X(3).	{ "YES" "NO" }		x x	Terminal status is requested (9763 Data Display Terminal); Terminal status is not requested			x								x	x		
03 FILLER	PIC X(6).		x	x															
02 MDATA	PIC X.	{ "Y" "N" }	x x	- -	More-data function is used; is <u>not</u> used			x			x								
02 FILLER	PIC X(2).		x	x															
02 RLTH	PIC 9(8) COMP.	0≤n≤65535	x	-	Maximum expected message length, in accordance with MDATA				*										
02 ROUTN	PIC 9(4) COMP.		x	x				x											
02 ROUTL	PIC X(8) OCCURS 8 TIMES		x	x				x											
02 FILLER	PIC X(12).		x	x															

1) All user entries are optional, some only conditionally so, however.

2) x Entered by user

+ Entered by DCAM (feedback information)

* Entered either by user or by DCAM (feedback information),
depending on application



Applies only to DCAM(NEA) transport service applications; for DCAM(ISO) transport service applications dummy parameter only.

Format of the distribution structure				Used by											
Level numbers, record names and data names	PICTURE clause	Parameter input 1)	Meaning	Y O C P E N S E N	Y O C L O N S C O N	Y O C R E L L O G	Y S E J L O G	Y C H A L O G E	Y I N Q U I T	Y P E R B I D	Y F O R D	Y S E N D	Y R E C E I V E	Y R E C E I V E	
01 VTLG-NAME			Record name												
02 CODEIDEN	PIC X(8).	structure-name	Symbolic name of structure		x			x		x					
02 CODELN	PIC 9(4) COMP.	(0<length <9)	Length of distribution code(s)		x										
02 CODEPOS	PIC 9(4) COMP.	(0<position <256)	Position of code in message		x										
02 CODEIND	PIC X.	code-indicator	Indicates implicit distribution code		x										
02 FILLER	PIC X.														
02 CODEGRNO	PIC 9(4) COMP.	(0<quantity <9)	Number of groups		x			x		x					
02 CODEGROUP OCCURS 8 TIMES.			Distribution code group(s)												
03 CODENAM	PIC X(4).	group-name	Symbolic name of group		x			x		x					
03 CODENO	PIC 9(4) COMP.	(0<amount <9)	Number of codes in group		x			x		x					
03 CODEVAL	PIC X(8) OCCURS 8 TIMES.	codes	Code values		x			x		x					

1) All user entries are mandatory.

Format of the wait structure					Used by														
Level numbers, record names and data names	PICTURE clause	Parameter	ISO	NEA	Meaning	Y O P E N	Y C L O S E	Y O P E N	Y C L O S E	Y R E S O L U T I O N	Y S T A T I C S	Y C H I Q U I R E	Y P R O C E D U R E	Y F R E Q U E N C Y	Y S E R I A L	Y R E C E I V E	Y R E S E T	Y W A I T	
01 WAIT-NAME.			x	x	Record name														
02 LIFETIM	PIC 9(5) COMP.	{1≤n≤42300} {60}	x	x	Wait time in seconds														x
02 FILLER	PIC X(12).		x	x															
02 DCAM-RETURN- INFO.			x	x	Return information														
03 RETCODE.			x	x	Return code														
04 WAIT-RC.			x	x															
05 RRS.	PIC 9(4) COMP.		x	x	Return item														+
05 RSI.	PIC 9(4) COMP.		x	x	Return item														+
04 EREIGNIS	PIC 9.		x	x	Event														+
04 FILLER	PIC X(3).		x	x															

6.2 CALL formats

CALL	USING	Function	Meaning
YCHANGE	APP-NAME, CONN-NAME, VTLG-NAME.	Connection	Change the characteristics of connection already established
YCLOSE	APP-NAME.	Existence	Close DCAM application
YCLSCON	APP-NAME, CONN-NAME.	Connection	Cancel request, or close virtual connection
YFORBID	APP-NAME, VT-NAME.	Data communication	Cancel assignment of a distribution code name (DISNAME) to distribution code group
YINQUIRE	FUNKTION, APP-NAME, CONN-NAME, BEREICH.	Existence Connection	Query information on applications and virtual connections
YOPEN	APP-NAME [,DCAMVER].	Existence	Open a DCAM application
YOPNCON	APP-NAME, CONN-NAME, BEF-NAME [,VTLG-NAME] [,BEREICH].	Connection	Establish a virtual connection
YPERMIT	APP-NAME, VT-NAME, VTLG-NAME, CG-NAME.	Data communication	Assign distribution code name (DISNAME) to a distribution code group
YRECEIVE	APP-NAME, CONN-NAME, BEF-NAME, [VTSUCB], BEREICH1, [,FHS-MAIN-PAR] [[,BEREICH-2]].	Data communication	Receive a message, express message or transport acknowledgment
YREJLOG	APP-NAME, CONN-NAME.	Connection	Reject connection request

CALL	USING	Function	Meaning
YRESET	APP-NAME, BEF-NAME, CONN-NAME.	Data communication	Change CS/CA state of a connection
YSEND	APP-NAME, CONN-NAME, BEF-NAME, [VTSUCB], BEREICH1, [,FHS-MAIN-PAR] [[,BEREICH2]].	Data communication	Send a message or express message
YSETLOG	APP-NAME, BEF-NAME.	Existence	Change the state of an application
YWAIT	WAIT-NAME, APP-NAME, CONN-NAME, BEF-NAME, [,FHS-MAIN-PAR] [[,BEREICH2]].	Asynchr. processing	Wait for occurrence of an event

6.3 Limit values for asynchronous calls

In DCAM the following limit values apply for the number of simultaneously open asynchronous calls:

- 128 YOPNCON (ACQUIRE) per application.
- 8 YOPNCON (ACCEPT, ANY) per application.
- 8 YOPNCON (ACCEPT, SPEC) per application.
- 8 YRECEIVE (SPEC) per connection.
- 8 YRECEIVE (ANY) per task of each application.

At the same time, the following limit values apply to DCAM COBOL programs:

Per task a maximum of 24 asynchronous calls are possible simultaneously, with at most 16 YOPNCON.

Asynchronous calls are:

- YOPNCON with SYN="NO".
- YRECEIVE with SYN="NO".
- YSEND after a return code "Waiting for GO" until arrival of the GO signal.

6.4 Feedback Information

Following each CALL, feedback information is generated and returned by DCAM to the user. The information is stored

- in the application structure after the calls:
YOPEN, YCLOSE, YCLSCON, YCHANGE, YINQUIRE, YPERMIT, YFORBID,
YREJLOG
- in the instruction structure after the calls:
YOPNCON, YSEND, YRECEIVE, YSETLOG, YRESET
- in the wait structure and in the instruction structure after the call: YWAIT

Note

Any return code that appears but is not described in the table is an original DCAM return code.

Feedback info.			Call													Meaning			
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J C T L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y R E N D	Y R E C E I V E	Y R E S E T	Y W A I T	Call was	Reason/explanation	
0			x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	executed	No error occurred
0	0		x															executed	Task is a primary task
0	4		x															executed	Task is a secondary task
0	1)	2)													x			executed	Additional information on message received or transport acknowledgment
0					x										x			accepted	The asynchronous job has been accepted
4	3)	2)			x		x		x	x				x	x			executed	Warning: call may not have been completely executed
8	3)		x	x	x	x	x	x	x	x	x	x	x	x	x			rejected	Current status of DCAM application prohibits execution
12	3)				x	x			x	x				x	x	x		rejected	Current status of partner prohibits execution
16	3)		x	x	x	x		x	x	x	x	x	x	x				rejected	Current status of access method (DCM) prohibits execution
20	3)				x	x	x	x	x	x	x	x						rejected	Illegal usage
24	3)		x		x	x	x	x	x	x	x	x	x	x				rejected	Erroneous data structures
28	3)		x	x	x	x	x	x	x	x	x	x	x	x	x	x		rejected	DCUS error
32			x	x	x	x		x	x	x	x	x	x					rejected	System error

- 1) Indicates data overflow
- 2) Acknowledgment or grouping indicator
- 3) Error code

Detailed explanations are to be found in the tables which follow.

Feedback info.			Call											Meaning				
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
0			x	x	x	x	x	x	x	x	x	x	x	x	x	x		Call successfully completed
0	0		x															Task is primary task } of the DCAM application Task is secondary task }
0	4		x															
0	0	.													x			Message does not exceed specified length of reception area
0	0	18														x		GO signal for messages has arrived
0	0	20														x		GO signal for express message has arrived
0	4	.													x			Message is longer than expected: remainder was discarded (TRUNC=YES)
0	8	.													x			Message is longer than expected: remainder has been kept (TRUNC=NO); length of remainder in ARECLN
		2													x			Transport acknowledgment - positive
		4													x			Transport acknowledgment - negative
		6													x			Element } Data grouping of normal message Subgroup } Group }
		8													x			
		10													x			
		12													x			Element } Data grouping of express message Subgroup } Group }
		14													x			
		16													x			
0	16														x			Message received, no VTSUCB used for this connection - VTSUCB is not returned
0	20														x			Message is longer than expected, remainder has been deleted, no VTSUCB used for this connection - VTSUCB is not returned
0	24														x			Message is longer than expected, remainder has been kept, no VTSUCB used for this connection - VTSUCB is not returned
0	64													x	x			Terminal status not yet available

Feedback info.			Call													Meaning	
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J C T L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T	
0	128														x		Terminal status complete
4					x		x	x			x	x			x		Call terminated with warning
4	4	*)													x		Inconsistent data coding (BINARY/SYSCODE) (*) as with RCD=0
4	8														x		No input data (NQ)
4	12				x		x										No connection request in queue
4	16				x										x		Call terminated because of timeout (Q and TOVAL)
4	20														x		Data lost through (system) timeout
4	24				x												Queued connection requests canceled because of (system) timeout
4	32									x							Connection message cut off
4	36													x	x		Error in message editing
4	40									x							Information has not been transferred completely, YINQUIRE aborted
4	44													x			Output truncated (edited message too long)
4	48								x								Invalid parameters entered for message editing (alteration not performed)
4	56													x	x		Warning in format handling by FHS, formatting completed (see FHS return code)
4	60													x	x		Warning in format handling by FHS, formatting not completed (see FHS return code)

Feedback info.			Call												Meaning			
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J E C T	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
8			x	x	x	x	x	x	x	x	x	x	x	x	x	x		Call rejected because of current status of DCAM application
8	4			x	x	x	x	x	x	x	x	x	x	x	x	x		DCAM application not opened by requesting task
8	8		x															DCAM application already opened (SHARE="NO" not possible)
8	12		x															DCAM application already opened and non-shareable (NSHARE)
8	16		x															Task has already opened this DCAM application
8	20		x															DCAM application already opened (VERIFY="PRM" not repeatable)
8	24		x															DCAM application not yet opened by primary task (VERIFY="SEC" not possible)
8	28		x	x	x					x				x	x			Primary task has closed the DCAM application
8	32		x		x						x							Warning: forced closure of DCAM application
8	36		x	x	x	x	x	x	x	x	x	x	x	x	x	x		Forced closure of DCAM application
8	40		x	x	x	x	x	x	x	x	x	x	x	x	x	x		Forced closure of DCAM application because of DCAM error
8	44		x	x						x				x	x			Forced closure of DCAM application because of termination of primary task
8	48													x	x			Secondary task has closed DCAM application
8	52										x							Distribution code name/distribution code group name already assigned (no further assignment allowed)
8	56		x															Invalid password (USEPW)
8	64				x										x			For this task too many calls of the same type have been given simultaneously (max. allowed: 8)
8	68		x															DCAM application inactive
8	72		x															Invalid password
8	76		x															Task has opened too many applications

Feedback info.			Call													Meaning		
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P E N C O N	Y C L O S E C O N	Y R E J E C T L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
8	80		x															Task has opened too many non-predefined applications
8	84		x															Invalid RDF password
8	88		x															Application already opened by another subsystem
8	92		x															Too many non-predefined applications have been opened
8	96			x														Too many connections have been established for non-predefined applications
8	100		x															The DCAM version DCAMVER of the secondary task does not correspond to that of the primary task.

Feedback info.			Call												Meaning			
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J E C T	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
12					x	x			x	x				x	x	x		Call rejected because of current status of partner
12	4						x			x	x			x	x	x		Partner is not connected to DCAM application
12	4				x													Partner is already connected to DCAM application
12	8				x													Partner has issued connection request (ACCP="NO" not possible)
12	12				x													Distribution structure not specified (DCAM application with DISCO)
12	16				x										x	x		Connection closed down by user call or YCLSCON
12	20																x	Waiting for GO signal terminated because of connection clear-down
12	20													x	x	x		Forced interruption of the connection or clear-down by the partner
12	24													x	x	x		Connection placed in CS state by another task (SPEC="YES" not possible from this task)
12	28														x			Connection is in CA state (SPEC="YES" not possible)
12	32				x									x				Message is too long (see MAXLN)
12	36				x													Specified position of distribution code (CODEPOS) greater than max. message length (MAXLN)
12	40													x				Transport acknowledgment expected previously has still not arrived for the reentered sequence number (SEQNO)
12	44				x													Illegal combination of data transmission protocol (DEPROT) and message editing (EDIT)
12	48				x													System timeout for connection request

Feedback info.			Call												Meaning			
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y P L S C O N	Y R E J L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
12	52				x													Partner cannot be reached (e.g. DCAM application not opened)
12	56				x													Partner presently not processing connection requests (application in STOP state)
12	60				x													Partner does not process connection requests (application with NLOGON attribute)
12	64				x													Invalid password
12	68				x													Request rejected by partner (no acceptance, or YREJLOG specified)
12	72									x								Invalid LID
12	76												x					Express message not transmittable when EDIT=SYS is set
12	80				x													Partner scheduled for other application
12	84				x													After connection was established, it was cleared again by partner or system
12	88				x													Request rejected by partner. Syntax error in connection message
12	92				x													Request rejected by partner Unknown authorization name 1)
12	96				x													Request rejected by partner. Duplicate application already opened 1)
12	100																	Request rejected by partner. Invalid password 1)

1) This information is returned when a connection request to system application "\$CONSOLE" has been rejected by the latter.

Feedback info.			Call													Meaning		
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
12	104														x	x		Connection cleared on account of partner error
12	108													x				Express message not permitted, because not agreed with partner
12	112				x													Fatal UCON error 1)
12	116				x													Invalid application name: first character not \$ 1)
12	120				x													Invalid processor (not own processor) 1)
12	124				x													Could not create task for verification 1)
12	128				x													Invalid CID for this OP-ID 1)
12	132				x													Internal UCON error (authentication not possible) 1)
12	136				x													No entry available in ECRNAM table 1)
12	140				x													DCAM version is less than 10 1)
12	144				x													Not a chipcard terminal 1)
12	148				x													Protocol inconsistent 1)
12	152				x													Chipcard subsystem not available 1)
12	156				x													Error in KVP protocol 1)

1) This information is returned when a connection request to system application "\$CONSOLE" has been rejected by the latter.

Feedback info.			Call											Meaning			
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T	
16	4	x	x	x				x	x		x		x	x	x	x	Lack of DCM memory space
16	4	4															BCAM: Attempt input/output later
16	4	8															BCAM: Resource bottleneck
16	4	12												x			Waiting for GO signal
16	4	32															DCAM: No control block available
16	4	36															DCAM: No space in table for ID entry
16	4	40															DCAM: No memory
16	4	44															DCAM: Bottleneck in partner system
16	4	48			x												DCAM: Too many DCAM events not fetched; only possible with YOPNCON
16	12		x	x	x	x	x	x	x	x	x	x	x	x	x	x	Termination of the DCM
16	16		x	x	x	x	x	x	x	x	x	x	x	x	x	x	DCM is not active
16	20		x	x	x	x	x	x	x	x	x	x	x	x	x	x	DCM error
16	20	4	x	x	x	x	x	x	x	x	x	x	x	x	x	x	BCAM: unknown BCINF return code
16	20	8	x	x	x	x	x	x	x	x	x	x	x	x	x	x	BCAM: unknown APINF return code
16	20	12	x	x	x	x	x	x	x	x	x	x	x	x	x	x	BCAM: unknown STINF return code
16	20	16	x	x	x	x	x	x	x	x	x	x	x	x	x	x	BCAM: unknown return code
16	24		x	x	x	x	x	x	x	x	x	x	x	x	x	x	DCAM currently not available

Feedback info.			Call												Meaning			
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y P L S C O N	Y R E J L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
20	4				x	x	x	x	x	x	x	x						Illegal use of call
20	4				x	x	x	x	x	x	x							Call cannot be issued from a secondary task
20	8											x	x					Call is not legal for non-shareable applications (NSHARE) nor for shareable ones (SHARE) not using distribution codes (NDISCO)
20	12				x			x										Not possible to accept a request or to place the application in the requested state as this application does not process requests (NLOGON)
20	16				x													Application is not authorized to issue the call or to use the instruction operands specified
20	20													x	x			Only when both TRANSF are set to "FOR" is FHS=YES executed
20	24													x	x			FHS="YES" not permitted (FHS module is missing)

Feedback info.			Call											Meaning				
RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
24			x		x	x	x		x	x	x	x	x					Call rejected because of invalid entry in data structures
24	8									x								Area too short
24	16				x													Error in distribution structure (distribution structure missing)
24	24		x															Error in application structure (APP-NAME) (application name (APPNAME) cannot be found)
24	28		x															Error in application structure (APP-NAME) (distribution code name (DISNAME) cannot be found)
24	32				x			x	x									Error in connection structure (partner name (PTNNAME) cannot be found)
24	44				x					x	x		x	x				Specified area is errored
24	48		x		x													Invalid application name
24	52		x								x	x						Invalid distribution code name (DISNAME)
24	56				x	x		x	x									Invalid partner name (PTNNAME)
24	60				x	x		x	x									Invalid processor name (PRONAME)
24	64												x					Value given as message sequence number (SEQNO) exceeds the permitted maximum
24	68									x			x	x				Area is too small (minimum length 8 bytes), or area length = 0
24	72										x							Number of codes in a group (CODENO) is not a permissible value
24	76										x							Symbolic name of code group (CODENAM) not contained in distribution structure

Feedback info.					Call												Meaning			
RSI	RRS	RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T		
		24	80				x			x					x	x				Non-recoverable error in message editing
		24	84												x					The FHS parameter structure is missing (specify DCAMVER ≥ V8.0)
		24	88												x	x				An error has occurred during format handling with FHS (see FHS return code in the FHS manual)
		24	92				x													Invalid distribution code length (CODELN)
		24	212									x								No VTSUCB available
		24	216												x	x				Connection does not work with VTSUCB (EDIT=USER)
		24	220												x	x				Invalid VTSUCB address
		24	224				x													Inconsistency between ROUTL and ROUTN
		24	228				x													Invalid ROUTL address

Feedback info.					Call													Meaning			
RSI	RRS	RCD	ECD	IND	Y O P E N	Y C L O S E	Y O P N C O N	Y C L S C O N	Y R E J L O G	Y S E T L O G	Y C H A N G E	Y I N Q U I R E	Y P E R M I T	Y F O R B I D	Y S E N D	Y R E C E I V E	Y R E S E T	Y W A I T			
		28			x	x	x	x	x	x	x	x	x	x	x	x	x	x		DCUS error	
		28	4		x															Limit for simultaneously opened applications exceeded	
		28	8				x													Limit for simultaneously active connections exceeded	
		28	12				x													Limit for simultaneously usable distribution structures exceeded	
		28	16				x						x							Limit for simultaneously usable distribution code groups exceeded	
		28	20												x	x				YRECEIVE (FHS=YES) was called before YSEND (FHS=YES), or the limit value for FHS parameter structures was exceeded	
		28	24		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x		It was not possible to reload DCUS module
		28	28												x					It was not possible to create common memory pool (no contiguous memory area)	
		28	32		x										x					No free memory space assigned	
		28	44				x										x			Area for CCBs exhausted	
		28	48				x						x				x			Area for RPBs exhausted	
		32			x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Messages 1) of this category should be reported as system errors. Please be sure to provide adequate documentation!	
		36			x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	Call not executed, either because DCAM subsystem was not loaded or because it was in HOLD/DELETE status and the task in question had not yet issued any DCAM calls.	
28	36																		x	Event has not occurred (either no asynchronous CALL given or the wait time (LIFETIM) has expired)	
28	40																			x	No DCAM event (e.g. foreign POSSIG)

Note

RCD=204 and ECD=0...255 are return codes which occur in conjunction with system exits; notify system administration (YOPEN, YOPNCON).

- 1) The same applies for entries which are not listed in this index.
- 2) The default values of module YDDCU004 are:
 - 10 applications
 - 20 connections
 - 10 distribution structures (different CODEIDENs)
 - 10 distribution code groups (different CODENAMs)
 - 20 FHS operand structures

These values can be increased by regenerating module YDDCUM04. This is accomplished via the ASSEMBLER macro call YDDCUM04. The setting can only be made by the system administrator. The value is then identical for all users.

Name	Operation	Operands
	YDDCUM04	C, xx, yy, zz, uu, vv

Operands

- C** must always be specified
- xx** limit for simultaneously opened applications ($0 < xx < 100$); default value = 50
- yy** limit for simultaneously active connections ($0 < yy < 100$); default value = 100
- zz** limit for simultaneously usable structure names (CODEIDEN) of distribution structures ($0 < zz < 100$); default value = 10
- uu** limit for simultaneously usable group names (CODENAM) of distribution code groups ($0 < uu < 100$); default value = 10
- vv** limit for simultaneously usable FHS operand structures (FHS-MAIN-PAR, $0 < vv < 100$); default value = 100

Note that system limits can be set to values < 100 , in which case the settings for xx and yy may be in excess of the maximum permissible system limits (BCAM). If this happens, the system administrator should change the system limits accordingly.

The source code of module YDDCU004 consists of two lines:

```
YDDCUM04 ...
END
```

The source code must be assembled. The object module must be placed in the task library (\$TSOS.SYSLIB.DCAM) of the DCAM COBOL module using LMS.

6.5 Name assignment commands

Two commands are available for name assignment. They are supervisor calls to the BS2000 Control System and are described in the "Control System Command Language" manual. This appendix will present a short functional description of them and show their formats.

1. APPLICATION command

The APPLICATION command enables a DCAM user to store or delete information about a DCAM application in a task-specific table, the communication link table (CLT).

When the DCAM application is opened by a DCAM application program, the values in this table replace the corresponding entries in the application control block ACB (ASSEMBLER) or in the application structure (COBOL). The CLT entry is linked to this program area by means of a link name (LINK) which must be specified both in the command and in the program.

Bear in mind that the /APPLICATION command can be used only when the DCAM subsystem has been loaded successfully. Note, too, that the DCAM subsystem status cannot be HOLD/DELETE when this command is used. If a task successfully issued a DCAM command or a DCAM call before entering HOLD/DELETE, it can work with DCAM until the task is ended, despite a /HOLD subsystem or /DELETE subsystem (also applicable to %).

Format

Name	Operation	Operands
	<pre>{APPLICATION} {APPL}</pre>	<pre>[application-name] ,LINK=link-name [,DISNAME=distribution-name] [,USEPASS=password1] [,USEPW=password2] [,LOGPASS=password3]</pre>

2. CONNECTION command

The CONNECTION command enables a DCAM user to store or delete information about a virtual connection in a task-specific table, the communication link table (CLT).

When the connection is being established, the values in this table replace or supplement the corresponding entries in the connection control block CCB (ASSEMBLER) or in the connection structure (COBOL). The CLT entry is linked to this program area by means of a link name (LINK) which must be specified both in the command and in the program.

Bear in mind that the /CONNECTION command can be used only when the DCAM subsystem has been loaded successfully. Note, too, that the DCAM subsystem status cannot be HOLD/DELETE when this command is used. If a task successfully issued a DCAM command or a DCAM call before entering HOLD/DELETE, it can work with DCAM until the task is ended, despite a /HOLD subsystem or /DELETE subsystem (also applicable to %).

Format

Name	Operation	Operands
	{ CONNECTION } { CONN }	[partner-name] ,LINK=link-name [,PRONAME=processor-name] [,USERFLD=user-item] [,PASSWORD=password]

These commands can also be entered in SDF syntax as shown below (for SDF Syntax, see the "User Commands (SDF Format)" manual).

REMOVE-DCAM-APPLICATION-LINK

Deletes information about a DCAM application from the CLT.

Format

REMOVE-DCAM-APPLICATION-LINK
LINK-NAME = <name 1..8>

REMOVE-DCAM-CONNECTION-LINK

Deletes information about a DCAM application from the CLT.

Format

REMOVE-DCAM-CONNECTION-LINK
LINK-NAME = <name 1..8>

SET-DCAM-APPLICATION-LINK

Stores information about a DCAM application in the CLT. This information is then used instead of the corresponding specifications in the program when the DCAM application is opened.

Format

```
SET-DCAM-APPLICATION-LINK
```

```
LINK-NAME = <name 1..8>
```

```
,APPLICATION-NAME = *BY-PROGRAM / <name 1..8>
```

```
,DISTRIBUTION-NAME = *BY-PROGRAM / <name 1..8>
```

```
,PROTECTION = PARAMETERS(...)
```

```
  PARAMETERS(...)
```

```
    CONNECTION-PASSWORD = BY-PROGRAM / <c-string 1..4> /  
                          <x-string 1..8> / SECRET
```

```
  ,SHARE-PASSWORD = BY-PROGRAM / <c-string 1..4> / <x-string 1..8> /  
                    SECRET
```

```
,SHARE-PASSWORD = BY-PROGRAM / <c-string 1..4> / <x-string 1..8> / SECRET
```

SET-DCAM-CONNECTION-LINK

Stores information about a DCAM connection in the CLT. This information is then used instead of the corresponding specifications in the program when the DCAM connection is established.

Format

```
SET-DCAM-CONNECTION-LINK
```

```
LINK-NAME = <name 1..8>
```

```
,PARTNER-ADDRESS = PARAMETERS(...)
```

```
  PARAMETERS(...)
```

```
    PARTNER-NAME = *BY-PROGRAM / <name 1..8>
```

```
    ,PROCESSOR-NAME = *BY-PROGRAM / <name 1..8>
```

```
,CONNECTION-PASSWORD = BY-PROGRAM / <c-string 1..4> / <x-string 1..8> /  
  SECRET
```

```
,USER-DATA = BY-PROGRAM / <c-string 1..4> / <x-string 1..8>
```

7 Manual supplements

This chapter is an update for the present manual valid for DCAM V13.3.

7.1 Feedback information

Various additions and corrections:

[Page 44](#): correction of 2nd paragraph under supplying:

02 TEXT PIC ... data field

[Page 179](#): additional meaning for ECD=4:

With YRECEIVE and EDITIN=PHYS, the user area is possibly too small to take the message header.

[Page 182](#): meaning for 08 88:

The application has already been opened on another (virtual) host.

[Page 184](#): meaning for 12 65:

With YOPNCON: the partner system has rejected the connection without giving a reason.

[Page 184/185](#): correction to footnote 1):

These messages are output if a connection setup request to the "\$CONSOLE" system application is rejected by it and the syntax of the connection message corresponds to a logon as logical console with generated authorization name. \$CONSOLE behaves differently for logon of a logical console with a dynamic authorization name.

Detailed information on the format of the connection message can be found in the manual "Introduction to System Administration".

[Page 188](#): correction to return code 24 48:

Invalid application or processor name

Page 191: Note 2)

2) The default values of the YDDCU004 module are:

50 Applications

100 Connections

10 Distribution structures (different CODEIDEN)

10 Distribution code groups (different CODENAM)

100 FHS operand structures

7.2 Commands for name assignment

Page 192: additional operand for APPLICATION command:

[,HOSTNAME=processorname]

processorname can be used to specify the name of a virtual host in which the application is to be opened.

Page 195: additional operand for SET-DCAM-APPLIC command:

,HOST-NAME=*PROGRAM/<name 1..8>

<name 1..8> can be used to specify the name of a virtual host in which the application is to be opened.

Page 195 - 196: general correction:

"*BY-PROGRAM" must always be used instead of "BY-PROGRAM"..

Glossary

CMX application

A communication application running on a SINIX or BS2000 computer and controlled by a CMX application program.

communication application

A facility for processing the messages exchanged by communication partners. It is addressed by the data communication system via its access point.

communication partners

Entities that maintain connections and exchange data with each other.

[communication] protocol

A description of the conditions and formats for transfer of information between equivalent functional layers in the data communication system.

communication computer

A computer designed specially for communication functions.

communication access method

The software that provides applications with an interface to the communication facility.

connection

A relationship between two communication partners that permits them to exchange data.

data unit

The quantity of data that can be passed to or received from DCAM with one call.

data communication system

A complex combination of hardware and software products that permits communication partners to exchange data in accordance with certain rules.

[DCAM] application

A communication application that is controlled by at least one DCAM application program.

[DCAM] application program

A program that uses the services of the DCAM access method; it controls one or more DCAM applications.

[DCAM] data transmission function

A DCAM function that is related to the transmission and reception of messages and acknowledgments.

[DCAM] event

A DCAM-specific event that can be used for coordination of certain operations in the data communication system. There is no specific time relationship between its arrival and the execution of the program (= asynchronous event).

[DCAM] existence function

A DCAM function that is related to the generation and cancellation of DCAM applications.

[DCAM] name assignment function

A DCAM function that permits the user to generate application programs independently of current parameter values, such as the DCAM application name, the partner name, etc.

[DCAM] connection function

A DCAM function that is related to the establishment and clearing down of connections.

express message

A message, with a restricted length, that is transmitted with a higher priority than normal messages.

format terminal

An operating mode of a virtual terminal where the message consists of a format (= entry form, screen mask).

line terminal

An operating mode of a virtual terminal where the message is structured in the form of lines.

logical terminal

--> virtual terminal

message

A logically related set of data that is to be transmitted to or received from a communication partner.

process

A facility for executing a program within a task.

shareable DCAM application

A DCAM application that can be used simultaneously by more than one task.

task

The carrier for processes. In BS2000, tasks are used, amongst other things, for execution of user jobs (e.g. batch job, interactive task) or for operation of (DCAM, UTM, TTX) applications (execution of all procedures specified between the BS2000 commands LOGON and LOGOFF).

terminal user

A person who uses a terminal to exchange data with a communication partner.

transport service

A service for the exchange of data between communication partners. The transport service initiates and monitors the transport of messages through the data communication system and manages connections.

transport acknowledgment

An event that provides information about the successful or unsuccessful execution of a data transfer.

virtual terminal (logical terminal)

A terminal model whose functions are mapped on the physical characteristics of various terminal types.

Related publications

DCAM (BS2000/OSD)

Program Interfaces

Reference Manual

DCAM (BS2000/OSD)

Macros

User Guide

FHS (BS2000/OSD)

User Guide

TIAM (BS2000/OSD)

User Guide

VTSU (BS2000/OSD)

User Guide

XHCS (BS2000/OSD)

Extended Host Code Support for BS2000/OSD

User Guide

CMX (BS2000)

Communication Method in BS2000

User Guide

COBOL85 (BS2000)

COBOL Compiler

User Guide

COBOL85 (BS2000)

COBOL Compiler

Reference Manual

BS2000/OSD

Commands

Volume 1-7

User Guide

BS2000/OSD

Executive Macros

User Guide

BS2000/OSD

Utility Routines

User Guide

BS2000/OSD

System Exits

User Guide

Ways to Open Communications

The ISO Reference Model in the Context of Communications

Brochure

Index

A

ACCPT 22
acknowledgment, logical 36
APP 48
APP-NAME 11
application
 changing state of 68
 closing 71
 non-shareable 15, 58
 open 57ff
 shareable 15, 60
application structure 11ff
application,status of 48
application-name 13
APPNAME 13
APPSTART 35
APPSTAT 48
asynchronous DCAM calls 53
audible alarm 25

B

backspace character 32
basic information, terminal 48, 96
BEF-NAME 17
BELL 25
BTERMINF 48, 96
BTI 48

C

call
 asynchronous 26
 synchronous 26
CALLs 51
calls,asynchronous 53
CG-NAME 49
character set 49, 94
CLT 31

- CNT 48
- code-indicator 40
- CODEGRNO 40
- CODEIDEN 39
- CODEIND 40
- CODELN 39
- CODENAM 41
- CODENO 41
- CODEPOS 40
- CODEVAL 41
- communication link table 31
- CONN-Name 27
- connection 27
 - change characteristics of 103
 - clearing down 106
 - description of 73
 - establish 72
 - setup 127
 - virtual 27
- connection clear-down, message on 131
- connection structure 27ff
- contents of the data item 44
- control characters 50
 - logical 36
- COUNTPTN 48, 100
- CS 21
- CS state 34
- CS/CA state, changing 119

D

- data
 - output protected 36
 - output unprotected 36
- data flow control 36
- data group 23
- data item
 - contents of 44
 - length of 44
- data structures 9
- data subgroup 23
- data transfer 35
- data-item 44
- DCAM version 49
- DCAMVER 49

default values 9
delete character 32
description of connection 73
DISCO 16
DISNAME 13
display terminal 49
distribution code
 length of 40
 position of 40
distribution code assignment, control of 120
distribution code group, name of 49
distribution code groups, number of 40
distribution code value 41
distribution codes 16, 62
distribution structure 39ff
distribution-name 13, 49

E

EDIT 34
element 23
entries
 query connection 89
 query partner 89
EREIGNIS 43
event 124
EXPRESS 23
EXTND 36

F

feedback field 43
FHS 25
format handling 25
format terminal 31, 33
function code, logical 32
FUNKTION 48

G

GETBS 32
GETFC 32
GO notification 129
GO-Signal 36
GOSIGNAL 43
group-name 41, 49

H

hardcopy 33
HCOPY 33
HOM 33

I

instruction execution
 asynchronous 124
 synchronous 123
instruction structure 17ff
ISO 17

L

LACK 36
LAENGE 44
LCASE 32
length of data item 44
LETTER 43
LIFETIM 43
line terminal 31, 33
lines, logical 33
LINK 13, 31
link-name 13, 31
linkage to a predefined connection 87
LINKMOD 14, 31
LOGC 36
logical acknowledgment 36
logical control characters 36
logical function code 32
logical lines 33
logical terminal 31
LOGON 16
LOGPASS 14
LOSCON 43, 132
lowercase letters, convert 32

M

MAXLN 35
MCS 49
MDATA 37
message
 conversion 34
 non conversion 35
 receiving 111, 125
 sending 108

message editing 33, 34
message length
 maximum 35
 maximum expected 38
 physical 35
message on connection cleardown 131
MONCHARS 49, 94
more-data function 37

N

NOEVENT 43
NORMAL 23
number of codes 41
number of partners 48
number of routes 38

O

OPENED 43

P

partner
 any 21
 specific 21
partner characteristics 48, 92, 135
partner count 100
partner information 48
partner-name 30
PASSWORD 30
password 14, 30
peripherals 49, 98
PEROTERM 49, 98
POT 49
PROCEDURE DIVISION 51
processor-name 13, 30
PRONAME 13, 30
PTN 48
PTNCHAR 48
PTNCHARS 92
PTNNAME 30

Q

queue
 access to common receiver 112
 accessing the originator-oriented 117
 common receiver 16, 22
 originator-oriented 16, 21

R

- reception area 34
- request
 - accept from any partner 81
 - acceptance of 80
 - canceling 106
 - for connection 76
 - from a specific partner 83
 - rejection 102
- RETCODE 43
- return information 43
- RLTH 38
- route names 38
- routes, number of 38
- ROU TL 38
- ROUTN 38

S

- secondary opening 64
- SEQNO 21
- sequence number 21
- SHARE 15
- SIGNAL 36
- single items 48
- SPEC 21
- standard areas 44ff
- START state 22, 68
- status of application 48
- STOP state 22, 68
- STP 22
- structure-name 39
- symbols used 5
- SYN 26
- SYSCODE 34

T

- TACK 16, 24
- task
 - arbitrary 15
 - primary 15
 - secondary 15
- terminal
 - basic information 48, 96
 - format 31, 33
 - line 31, 33

logical 31
virtual 31, 33
terminal status, check 37
TERMSTAT 37
TOP 48
TOPLOGON 48
TOVAL 20
TRANSF 31, 33
transfer areas 9
transport acknowledgment 16
 positive 24
 receiving 111
TRUNC 24, 34

U

USEPASS 14
USEPW 15

V

VERIFY 15
virtual terminal 31, 33
VT-NAME 49
VTLG-NAME 39
VTSUCB 26

W

wait structure 42
wait time 53
WAIT-NAME 42

Y

YCHANGE 103
YCLSCON 106
YFORBID 122
YINQUIRE 89ff
YOPEN 57ff
YOPNCON 72ff
YPERMIT 120
YRECEIVE 111ff
YREJLOG 102
YRESET 119
YSEND 108ff
YWAIT 125ff

Contents

1	Preface	1
1.1	The DCAM manual concept	2
1.2	Changes since the last version of the manual	4
1.3	Notational conventions	5
2	Characteristics of the DCAM COBOL interface	7
2.1	Summary	7
2.2	Description of data structures and transfer areas (WORKING-STORAGE SECTION)	9
2.2.1	General	9
2.2.2	Application structure (APP-NAME)	11
2.2.3	Instruction structure (BEF-NAME)	17
2.2.4	Connection structure (CONN-NAME)	27
2.2.5	Distribution structure (VTLG-NAME)	39
2.2.6	Wait structure (WAIT-NAME)	42
2.2.7	Additional transfer areas	44
2.2.7.1	Standard areas	44
2.2.7.2	Areas for the YOPNCON call	46
2.2.7.3	Single items	48
2.2.7.4	Areas with control characters	50
2.3	Description of the CALLs (PROCEDURE DIVISION)	51
2.3.1	Execution of (subroutine) calls	51
2.3.1.1	Wait times	53
2.3.1.2	Execution with asynchronous DCAM calls	53
3	Using the DCAM functions	55
3.1	CALLs relating to the existence of a DCAM application	57
3.1.1	Opening a DCAM application	57
3.1.1.1	Opening a non-shareable DCAM application	58
3.1.1.2	Primary opening of a shareable DCAM application	60
3.1.1.3	Primary opening - use of distribution codes	62
3.1.1.4	Secondary opening	64
3.1.1.5	Secondary opening - use of distribution codes	66
3.1.2	Changing the state of a DCAM application	68
3.1.3	Querying the state of a DCAM application	69
3.1.4	Closing a DCAM application	71
3.2	Connection-oriented CALLs	72

3.2.1	Establishing a connection	72
3.2.1.1	Definition of the connection to be established	73
3.2.1.2	Connection request	76
3.2.1.3	Accepting a request	80
3.2.1.4	Connection setup - use of distribution codes	85
3.2.1.5	Linkage to a predefined connection	87
3.2.2	Querying partner and connection entries	89
3.2.2.1	Partner information prior to connection setup (TOPLOGON)	90
3.2.2.2	Partner characteristics (PTNCHARS)	92
3.2.2.3	Display terminal and character sets (MONCHARS)	94
3.2.2.4	Basic terminal information (BTERMINF)	96
3.2.2.5	Peripherals (PEROTERM)	98
3.2.2.6	Partner count (COUNTPTN)	100
3.2.3	Rejecting a connection request	102
3.2.4	Changing the characteristics of a connection	103
3.2.5	Canceling a request	106
3.2.6	Clearing down a connection	106
3.3	Data communication-oriented CALLS	107
3.3.1	Sending a message	108
3.3.2	Receiving a message or transport acknowledgment	111
3.3.2.1	Accessing the common receiver queue	112
3.3.2.2	Accessing the originator-oriented queue	117
3.3.3	Combined sending and receiving	119
3.3.4	Changing the CS/CA state of a connection	119
3.3.5	Controlling distribution code assignment	120
3.3.5.1	Assigning distribution code names to distribution code groups	120
3.3.5.2	Canceling the assignment	122
3.4	CALLs for asynchronous processing	123
3.4.1	Waiting for termination of asynchronous DCAM jobs	124
3.4.1.1	Receiving messages	125
3.4.1.2	Setting up a connection	127
3.4.1.3	GO notification	129
3.4.1.4	Message on connection clear-down	131
4	Partner characteristics	135
5	Examples	141
5.1	DCAM(ISO) transport service	141
5.2	DCAM(NEA) transport service	144

6	Appendix	165
6.1	Format of the data structures	165
6.2	CALL formats	174
6.3	Limit values for asynchronous calls	176
6.4	Feedback Information	177
6.5	Name assignment commands	192
7	Manual Supplement	197
	Glossary	
	Related publications	
	Index	

DCAM (BS2000)

COBOL Calls

Valid for

DCAM V11.0A

With [Supplement chapter for DCAM V13.3A](#)

Comments... Suggestions... Corrections...

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Feel free to send us your comments by e-mail to:
manuals@ts.fujitsu.com

Certified documentation according to DIN EN ISO 9001:2000

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2000.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Copyright and Trademarks

Copyright © Fujitsu Technology Solutions GmbH 2010.

All rights reserved.

Delivery subject to availability; right of technical modifications reserved.

All hardware and software names used are trademarks of their respective manufacturers.



On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document is a new edition of an earlier manual for a product version which was released a considerable time ago in which changes have been made to the subject matter.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format ...@ts.fujitsu.com.

The Internet pages of Fujitsu Technology Solutions are available at [http://ts.fujitsu.com/...](http://ts.fujitsu.com/)