Network Generation

User's Guide

DPS7000/XTA
NOVASCALE 7000

Communications: FCP7

REFERENCE
47 A2 93UC 04

DPS7000/XTA
NOVASCALE 7000
Network Generation
User's Guide

Communications: FCP7

August 2002

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE
47 A2 93UC 04

Printed in France

Suggestions and criticisms concerning the form, content, and presentation of this book are invited. A form is provided at the end of this book for this purpose.

To order additional copies of this book or other Bull Technical Publications, you are invited to use the Ordering Form also provided at the end of this book.

Trademarks and Acknowledgements

Preface

Scope and Objectives

This manual discusses GCOS 7 Network Generation using the NETGEN utility. It also describes OCS Front End configuration.

Network Generation treats the objects which the user declares on the host DPS 7000 or DPS 7000/XTA (eXtended Twin Architecture) to configure the network. These objects are defined by directives which are then submitted to the NETGEN utility for compilation.

The directives for Network Generation are treated in categories of their function, namely:

- those for configuring the Network
- those for administrating the Network
- and those for establishing the Service of Correspondents.

Maintenance includes tuning communications applications by allowing for sizing parameter values for communications servers and user-defined queues.

Configuration of an OCS Front End treats the objects necessary to communicate with remote systems via the OCS Front End under OSI or TCP/IP.

Hardware Platform and GCOS 7 Release

This document may have generic references to a DPS 7000 hardware platform. If so, such references are applicable to all models of the following Bull large-system computers.

- DPS 7000 GCOS 7 V9, TS 9866
- DPS 7000/XTA GCOS 7 V10, TS 9910

For a few keywords, the default value depends on the DPS 7000 system model (XTA or not).

Bibliography

The following documents may be consulted for further information on the main topics referred to in this manual.

MainWay and MainWay 2600LE:

MainWay Overview 39 A2 14EB
MainWay 2600LE User’s Guide 39 A2 60AE
MainWay 2600LE HW, SW & Maintenance Guide 39 A2 61AE
MainWay 2600 Site Preparation Guide 39 A1 82RA
DNS-E System Generation Guide 39 A2 09EB
NGL Reference Manual 39 A1 10EB

Datanet and CNP 7:

DNS V4 System Generation 39 A2 22DN
CNS 7 System Generation 39 A2 40DM
Simple Generation Guide using FPG7 .... 39 A2 46DM

Transport and Session Servers:

VCAM-ISO Reference Manual (Part 1) 47 A2 60UC
VCAM-ISO Reference Manual (Part 2: primitives) 47 A2 61UC
VCAM-ISO User’s Guide 47 A2 62UC

Network Installation, Operation and Administration:

Getting started with your Telecommunications 47 A2 70UC
Network Overview and Concepts 47 A2 92UC
Network User Guide 47 A2 94UC
DSAC User’s Guide 47 A2 75UC
AUPI User’s Guide 47 A2 76UC

Service of Correspondents:

Transactional Intercommunication using XCP1 User’s Guide 47 A2 11UT
CPI-C/XCP2 User’s Guide 47 A2 14UT
TDS Administrator’s Guide 47 A2 20UT

DPS 7000 to XTA Evolution Guide:

V7000 Operator's Guide... *47 A2 74US*
V7000 Configuration and Maintenance Guide...*77A2 77US*
V7000 Software Installation and Activation Guide *77 A2 88US*
GCOS7 - System Installation Configuration and Updating Guide.......... *47 A2 23US*
GCOS7-V10 System Operator's Guide.. *47 A2 53US*
DPS 7000/XTA Interop 7 User's Guide.. *47 A2 91US*
GCOS 7 Console Messages Directory.. *47 A2 61UU*
GCOS 7 Messages and Return Codes Directory *47 A2 10UJ*

Other Documents:

INET Reference Manual *13 A2 12SM*
INET User Guide *13 A2 13SM*
MCS User's Guide *47 A2 32UC*
TCP/IP 7 End-User's Guide *47 A2 30US*
OPEN 7 Administrator's Reference Manual.. *47 A2 31US*
OPEN 7 Administrator's Guide *47 A2 32US*

Syntax Notation

The following notation conventions are used in this manual when describing the syntax of commands:

UPPERCASE	The keyword item must be coded exactly as shown.
Lowercase	Indicates a user-supplied parameter value. The symbolic name digitsn is used to represent a string of decimal digits of maximum length n.
[item]	An item within square brackets is optional.
{item 1}	
{item 2}	
{item 3}	A column of items within braces means that one value must be selected if the associated parameter is specified. The default value (if any) is underlined.
( )	Parentheses must be coded if they enclose more than one item.
. . .	An ellipsis indicates that the preceding item may be repeated one or more times.

GCOS 7 Documentation

The complete GCOS 7 document set is available on one CD-ROM. This product is known as CD-DOC. It is delivered with each system or software update, it is updated for each new GCOS 7 release. A WEB site is available to download documents which have been updated between 2 CD-ROM versions, its URL is indicated as a link on the CD-ROM itself.

Table of Contents

7. Improving the Network Description

8. OCS Front End Configuration

A. NETGEN Reports

B. Server Job Occurrence Reports

Table of Graphics

Figures

1. Introduction

1.1 Scope

To allow GCOS 7 applications to dialog with applications located in remote system, you must configure the network (to define the DPS 7000 or DPS 7000/XTA (eXtended Twin Architecture) view of the network). This view is defined using the NETGEN utility.

This manual describes the directives and parameters used to define the DPS 7000 or DPS 7000/XTA view of a network (and they're by enable GCOS 7 to communicate with remote systems in the network).

1.2 NETGEN

A DPS 7000 or DPS 7000/XTA communications environment is composed of:

- The local system (the DPS 7000 or DPS 7000/XTA)
- the Datanet front-ends and their associated FEPS servers,
- the ISL (Inter System Link) communications and CNP7s managed by the TNS server,
- the OCS Front Ends managed by the OCS servers,
- the remote systems accessed from GCOS 7 through FEPS, TNS, or OCS,
- administrative functions to control and tune the network,
- queue parameters for QMON and MCS parameters.

Chapters 2 through 7 discuss the NETGEN utility (used to describe the GCOS 7 view of a network).

1.3 OCS Front End Configuration

If communications are to be made via an OCS Front End, you must configure the OCS Front End.

OCS Front End configuration consists of using directives and parameters to describe the access to remote systems through OCS Front End (via ISO, ISO/DSA, and TCP/IP).

Chapter 8 describes OCS Front End configuration (for ISO, ISO/DSA, and TCP/IP).

1.4 Miscellaneous

Chapter 9 discusses network tuning.

Chapter 10 discusses network limits.

The Appendices gives examples of reports, terminal tables, MCS queues, and discuss migration from earlier releases of GCOS 7.

For introductory information on GCOS 7 communications, see the manual *Network Overview*.

2. Activating the NETGEN Utility

2.1 Purpose of NETGEN

Communications configurations are generated by the NETGEN alias NG (Network Generator) batch processor. It is started by the GCL command CRNETGEN.

The version of NETGEN delivered with the current release can only handle configurations which it itself creates and saves. This means that configurations generated under releases prior to this release must be regenerated before being installed under this release.

Generating the communications configurations involves the following steps:

- generating the standard and site-specific terminal configuration
- generating the basic network configuration
- and generating the directory configuration which comprises:
  - the dictionary of remote systems
  - the dictionary of correspondents.

Generating the directory is also called incremental generation. It allows you to add and/or suppress objects (remote systems or correspondents) without stopping the telecommunication servers.

Additional functions are provided when generating the configurations:

- **SAVING** the configurations in system or user-defined binary library members from where the configuration can be later retrieved through the **LOADING** function
- **ENABLING** the configurations which involves installing them to be fully accessible to all the communications servers
- **INCREMENTAL ENABLING** of the directory configuration which involves replacing the directory empty or otherwise currently running, by another directory updated and not empty, while the two other configurations are running
- **INCREMENTAL DISABLING** of the directory configuration which involves replacing the directory which is not empty and currently running, by an empty directory while the two other configurations are running.

2.2 Functions of NETGEN

NETGEN performs the following actions in whatever order, depending on the parameters specified:

CRNET=member: generates the basic network configuration from the description supplied by this member.

CRDIR=member: generates the directory configuration from the description supplied by this member.

CRTERM=member: generates the terminal configuration from the description supplied by this member and by that contained in the H_TERM of the SYS.HSLLIB library.

LOAD: retrieves all the configurations from binary library members and moves them into the permanent workspace (if enabling is requested) or into a temporary workspace.
Members may be:
- system members (known at GCOS 7 restart time) if LOAD=SYSTEM is specified
- private members by specifying LOAD=member.

SAVE: saves all the configurations from the permanent workspace (if enabling is requested) or from a temporary workspace into binary library members which can be:
- system members (known at GCOS 7 restart time) if SAVE=SYSTEM is specified
- private members by specifying SAVE=member.

ENABLE: moves all the generated or loaded configurations from the permanent workspace into the running workspace allowing them to be accessed by the communications servers. Enabling with *LOAD* or *CRNET=member* is effective only if the communications session is terminated. Incremental enabling with *CRNET=NO* and *CRDIR=member* can be performed while a communications session is in progress.

NOTE:
A generation with *SAVE=NO* and *ENABLE=0* produces output in a temporary workspace whose contents is lost when NETGEN terminates.

2.3 Overview of the Network Generation Process

The NETGEN processor can be viewed as the sequential processing of the following independent and optional functions.

The first processing involves compiling the network configuration descriptions according to the *CRNET*, *CRDIR* or *CRTERM* parameters, namely:

- analyze the syntax of directives
- if no error detected, perform global checks
- store the generated configurations in:
  – either the permanent workspace if ENABLE is requested
  – or a temporary workspace

Any error at this stage does not affect the running communications session and the job is aborted.

Subsequent processing continues according to the following options:

- If neither *SAVE* nor *ENABLE* is specified, the final report is issued and the processing ends here.
- If *LOAD* is requested, the configurations are fetched from the binary library and moved into:
  – either the permanent workspace if *ENABLE* is requested
  – or a temporary workspace in which case, either *SAVE* or *ENABLE* must be specified with values other than *NO*.
- If *SAVE* is requested, the contents of either the permanent workspace where ENABLE is requested or temporary workspace are moved into:
  – either the system library members *(SYS.DSACORE..NETGEN*)* if *SAVE=SYS*
  – or the user-defined library members if *SAVE=member*.
- If *ENABLE* is requested, the generated or loaded configurations are made available to the communications servers by moving them from the permanent workspace into the running workspace.

The charts overleaf illustrate the main NETGEN capabilities.

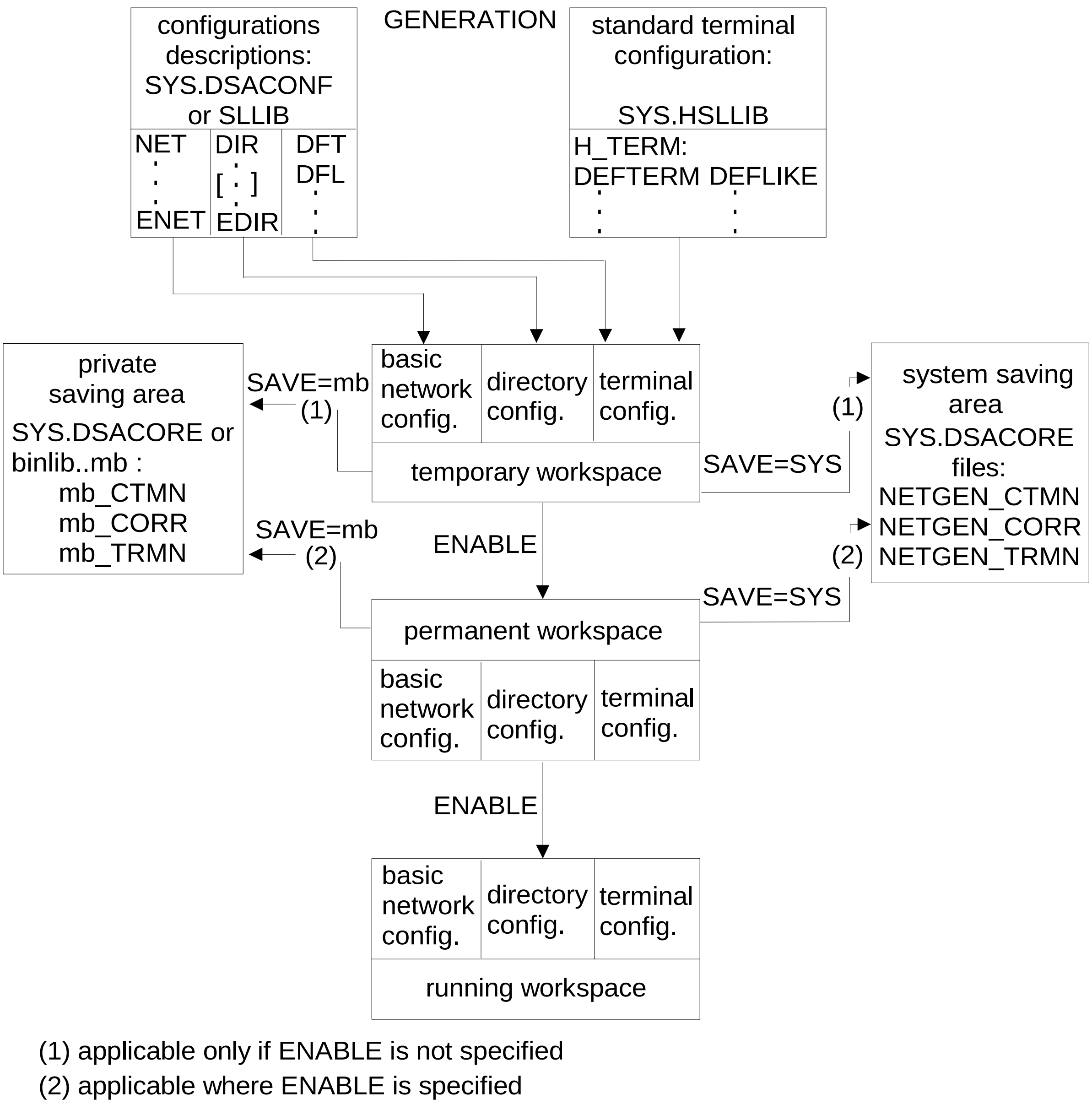
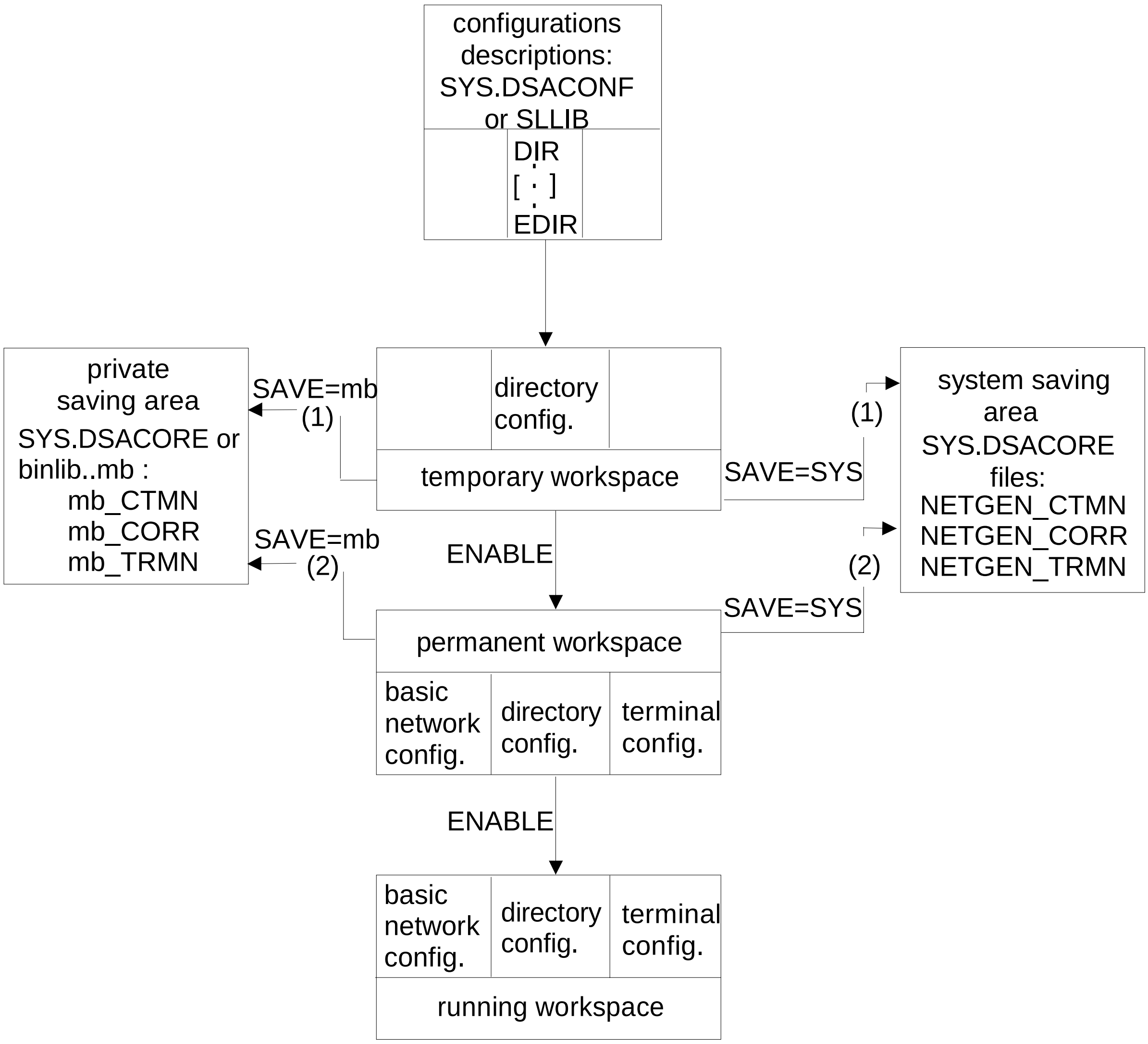


Figure 2-1. Flowchart of Network Generation

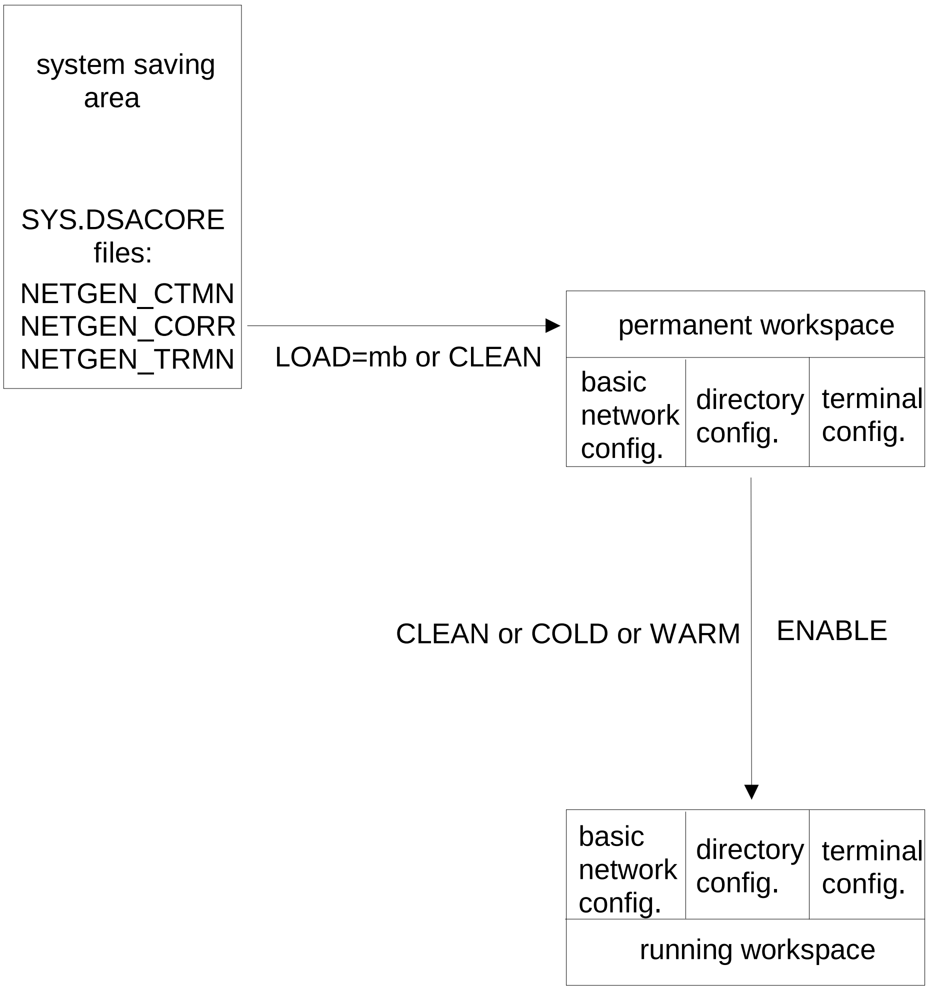


(1) applicable only if ENABLE is not specified

(2) if ENABLE is specified, all 3 configurations are moved; otherwise, only the basic network and terminal configurations are moved.

Figure 2-2. Flowchart of Incremental Generation

Figure 2-3. Flowchart of Private Loading



System Loading involves GCOS 7 restart.

Figure 2-4. Flowchart of System Loading

2.4 SRST Information

The SRST describes all the hardware components and their associated firmware within an installation, in terms of:

- their hierarchical structure, being the CPU, peripheral processors, peripherals, including the console operator
- and their identifiers such as CChh for the PSI of the Datanet, and EAhh for the ISL (Ethernet and/or FDDI) attachments.

In an SRST listing, the status of the hardware components is designated as follows:

- ***C*** for not connected
- ***F*** for (connected but) failed
- ***A*** for (connected and) available.

NOTE:

C means that the hardware component was not physically connected or not declared at firmware generation. When the DPS 7000 is initialized, the component is therefore unavailable for the GCOS 7 session.

Each time *saving* or *enabling* the generated configurations is requested, NETGEN checks that the LCTs declared in the basic network configuration description are consistent with the SRST contents.

Configurations *saved* for another system can be checked against the file(s) into which the SRST of the targeted system concerned has been copied.

CChh or EAhh (in the NAME parameter of the LCT directive) identifies the type of DPS 7000 controller used.

For each CChh and for each EAhh, NETGEN verifies both conditions in the SRST:

- firstly, if they do exist
- and, if they do, if there are sufficient CLnn's or LTnn's, respectively, defined for each of them.

If both conditions are not satisfied, a warning is given. TNS, FEPS, and OCS perform their own checks when initialized.

NOTE:

The term *CLnn* is the SRST entry for the CCLINK (Communications Link) associated with CChh.

The term *LTnn* is the SRST entry for the ELT (Ethernet Logical Terminator) or FLT (FDDI Logical Terminator) associated with EAhh.

2.5 Operational Environment

For a basic generation (that is, non-incremental generation), the ENABLE function which moves the permanent workspace into the running workspace requires that no communication session is *in progress*, namely:

- no communications server (TNS, FEPS, FECM, QMON, RAEH, or OCS) must be still active
- no network operator command should be executing
- no application using VCAM such as TDS transactions and MCS programs, must be executing.
- no OPEN7 running

The running workspace is reinitialized from the permanent workspace at each new GCOS 7 session on COLD or WARM restart.

However, the running workspace is deleted by a new GCOS 7 session with CLEAN restart. In this case, the subfiles NETGEN_CTMN, NETGEN_CORR and NETGEN_TRMN created by the last SAVE=SYS in the SYS.DSACORE library are reloaded in the permanent workspace depending on the response given to the following message:

```
NG03 DO YOU WANT TO LOAD PREVIOUSLY SAVED
TELECOMMUNICATION CONFIGURATIONS? (Y,N),DEFAULT = Y.
```

2.6 Typical Generation Scenarios

The functions provided by NETGEN allow a wide combination of scenarios, depending on how the network is to be administered. Three typical scenarios are treated below.

2.6.1 Initial Generation

When no previous generation exists, one must be created and enabled by NETGEN before any communications server is started, and then tested. CRNETGEN command can be executed at the system console with the following parameters:

CRNET=member, CRDIR=member, SLLIB=sl_library, ENABLE

The configuration active under previous releases can be regenerated through the following parameters:

CRNET=member, SLLIB=sl_library, ENABLE

2.6.2 Updating a Generation

If some communications sessions are currently running and if the network requires regeneration, the current generation must not be tampered with until the new generation has been fully tested and proved to be satisfactory.

The following steps are recommended:

Step A

Compile the updated configuration description using **CRNET=member, CRDIR=member** and **SAVE=member** to check the syntax and save the generated configurations without perturbing the current communications session. Should a syntax error be detected for any reason such as duplicate or missing objects, SAVE is discarded. Recompile until the generation is error-free and the NETGEN job completes normally.

Step B

The new generation is now ready to be tested. It is necessary to stop the communications session. Terminate the communication servers through the TTSVR network operator command before the new generation is made available to them. Verify that all servers are in the *enable* state using the DTSVR network operator command.

Step C

Submit the CRNETGEN command with **LOAD=member** and **ENABLE=1**. On normal completion of the job, the new generation is available for use by all the communications servers. It will be reloaded at the next GCOS 7 session on COLD or WARM restart.

Step D

Start a new communications session and whatever servers are needed through the STSVR network command. Try accessing systems described either in the basic network configuration or in the directory configuration by submitting UFT or DJP requests.

Step E

The behavior of the running communications servers can now be analyzed for any wrong or missing objects which were not detected in the preliminary compilation step A.

Step F

If the current configuration is to be modified once more, and if the error is in the basic network configuration, two solutions are possible:

- either repeat the entire process from Step A until the configuration is satisfactory
- or before returning to the Step A, revert to the configuration which was installed before the currently tested one. To do this, terminate the current communications session and submit the **CRNETGEN** command with **LOAD=SYSTEM** and **ENABLE=1**.

If the error is in the directory configuration, correct it and resubmit the CRNETGEN command with **CRNET=NO CRDIR=member SAVE=member ENABLE=1**.

Step G

When the current generation has proved satisfactory, the previous generation, if no longer required, can be overridden in the SYS.DSACORE library. To do this, submit the CRNETGEN command specifying **LOAD=member** and **SAVE=SYSTEM**. Once this is done, the new generation will be systematically loaded for all subsequent GCOS 7 sessions even on CLEAN restart.

NOTE:

If normal activity is terminated to allow for installing new or additional hardware and software equipment, the easiest way to create the new generation is:

1. to terminate the communications session and the servers as in Step B
2. to submit the CRNETGEN job specifying CRNET, SAVE and ENABLE, assuming an error-free compilation, thereby merging Steps A and C
3. and then, to proceed to Steps D and G.

2.6.3 Generation for a Remote System

When saving and/or enabling is requested, NETGEN checks if the local system name supplied by the SYSTEM directive matches the actual local DPS 7000 name and if some described hardware objects are actually registered in the local SRST table.

In some circumstances, these checks must be inhibited or modified for example, when preparing on a given dedicated DPS 7000, all the configurations for several remote DPS 7000s. This can be done without affecting the communication sessions of the local host by submitting a CRNETGEN command with:

CRNET=member CRDIR=member SAVE=member SRSTFILE=file

The effects of the SRSTFILE parameter are:

- the local system name supplied by the SYSTEM directive is not checked against the actual name of the DPS 7000 where the NETGEN is running
- the hardware objects names specified in the basic network configuration are matched against those registered in the remote SRST file.

The file specified in the SRSTFILE parameter must contain the SRST of the remote targeted DPS 7000 and may be allocated as follows where underlined values are imperative:

```
S: BUILD_FILE 'mysrst:media:devclass'
   UFAS    = SEQ,
   UNIT    = CYL,
   SIZE    = 1,
   MAXEXT  = 5,
   CISIZE  = 2100,
   RECSIZE = 512,
   RECFORM = V
   SILENT  = 0;
S:
```

NETGEN issues the following banner:

```
SRST OF THE MACHINE dec8    SAVED AT  time  date
```

where dec8 is up to eight decimal digits representing the internal identification of the remote targeted system.

The user must verify that the DPS 7000 identified by this number is consistent with the local system name specified in the basic network configuration description.

This file must have been built on the targeted remote DPS7 using the H_SVSRST utility:

```
S: EXEC_PG H_SVSRST,
           LIB   = SYS.HLMLIB,
           FILE1 = H_SRST,
           ASG1  = (mysrst:media:devclass);
S:
```

Deallocating then allocating the SRST file, and invoking the H_SVRST utility can also be done using the following command:

```
EJ SAVE_SRST,,SYS.HSLLIB
   VL=(mysrst,MD=media,DVC=devclass[,FORCE={Y|N}]);
```

where mysrst is the name of the SRST file and FORCE=Y deallocates the file specified.

The SRST files and all the generated configurations can be moved site to site using the UFT utility. For instance:

```
                         {CTMN}                   {member} {CTMN}
EFTR SYS.DSACORE..member_{CORR} $site:SYS.DSACORE..{      }_{CORR}
                         {TRMN}                   {NETGEN} {TRMN}
```

On the remote systems,the remotely prepared configurations have to be installed using respectively one of the two following CRNETGEN commands:

```
                { member, SAVE=SYSTEM,}
CRNETGEN  LOAD= {                     } ENABLE=1 ;
                { SYSTEM,             }
```

2.7 Executing the Network Generation

Syntax

```
{ CRNG | CRNETGEN | NG       }
{                            }
{ CREATE_NETWORK_GENERATION }

[           { NO  [,{ TERM | CRTERM } = NO ]               } ]
[ { NET   } {                                              } ]
[,{       }={               [ { TERM   } { NO            } ] } ]
[ { CRNET } { member-name [,{        }={ STANDARD | STD } ] } ]
[           {               [ { CRTERM } { member-name    } ] } ]

[ { DIR   }   { NO          } ]    [          { SYS.DSACONF } ]
[,{       } = {             } ]    [,SLLIB = {             } ]
[ { CRDIR }   { member-name } ]    [          { lib-name-78 } ]

[            { NO           } ]    [            { NO           } ]
[            {              } ]    [            {              } ]
[ { LD   }   { SYSTEM | SYS } ]    [ { SV   }   { SYSTEM | SYS } ]
[,{      } = {              } ]    [,{      } = {              } ]
[ { LOAD }   { member-name  } ]    [ { SAVE }   { member-name  } ]
[            {              } ]    [            {              } ]
[            { full-mb-nm   } ]    [            { full-mb-nm   } ]

[ { ENBL   }   { 0 } ]             [,QFILE = file-literal ]
[,{        } = {   } ]
[ { ENABLE }   { 1 } ]             [,SRSTFILE = file-literal ] ;
```

Parameter Description

NET alias **CRNET**

Specifies the name of the member in the input source library introduced by SLLIB which contains the set of commands to be processed for creation of the basic communication network configuration. The first directive other than COMM must be the directives NET. The last directive other than COMM may be ENET. The member cannot contain directives specific to the directory such as SDOM, and correspondents directives. See Section 5.

CRNET=NO: ***Default***
Specifies that no basic communications network configuration is to be created.

DIR alias **CRDIR**	Specifies the name of the member in the input source library introduced by SLLIB which contains the set of commands to be processed for creating the correspondents, remote systems and subdomains of the Directory configuration. The first directive other than COMM must be the directive DIR. The last directive other than COMM may be EDIR. This member cannot contain directives specific to the basic configuration such as the local system, passthroughs and neighbors. See Section 5. No directives other than COMMs between DIR and EDIR directives means an empty directory configuration description. Generating and enabling such a description results in a valid empty directory. When specified with CRNET=NO, it denotes an incremental generation. In this case, saving or enabling requires that a valid set of configurations has been previously installed in the ***running*** workspace. The incremental generation and enabling of an empty directory configuration description will disable an already enabled and active directory. *CRDIR=NO:* ***Default*** Specifies that no directory configuration is to be created. When specified with CRNET=member-name, and SAVE or ENABLE, an ***empty*** directory configuration is created which is initialized for supporting a possible further incremental generation.
TERM alias **CRTERM**	***Effective only when specified with CRNET=member-name.*** Requests the MAINTAIN_TERMINAL processor to perform the following actions: *CRTERM=NO:* ***Default*** It normally specifies that the terminal table is not to be created except when it does not exist in the permanent workspace and it is needed by the running context. In this case, NO is equivalent to STANDARD. *CRTERM=STANDARD:* The terminal table will be created from the standard-defined directives contained in the member H_TERM in the library SYS.HSLLIB. *CRTERM=name31:* Functions like STANDARD, in addition to processing user-defined directives in the named member of the library introduced by SLLIB.

SLLIB	**Meaningless when *CRNET=NO* and *CRDIR=NO*, and *CRTERM=NO* or *CRTERM=STD*.** *SLLIB=SYS.DSACONF:* ***Default*** *SLLIB=file78:* Specifies the name of the input library containing the sources to be processed by CRNET, CRDIR and CRTERM. It cannot be TEMPRY.
LD alias **LOAD**	Specifies if the generation is to be loaded and from where. Loading involves moving previously saved configurations from a specified binary library into a workspace selected from ENABLE: – **permanent workspace** if ENABLE=1 – **temporary workspace** if ENABLE=0. **Since loading and creation are mutually exclusive, loading is effective when *CRNET=NO* and *CRDIR=NO* and *CRTERM=NO*.** LOAD is effective when specified with SAVE and/or ENABLE, and always involves the whole configuration set being: basic + directory + terminal. In addition all these components must have been saved by the same run occurrence of the CRNETGEN command. *LOAD=NO:* ***Default*** No loading from binary library is requested. *LOAD=SYSTEM:* Specifies that the set of configurations to be loaded is the most recently saved with SAVE=SYSTEM. It requires SAVE=NO and ENABLE=1. This action is similar to that performed at GCOS 7 initialization time on RESTART CLEAN. *LOAD=file78* of the format: *library-name..member-name*. It cannot be TEMPRY. Specifies that the set of configurations to be loaded is one previously saved and where it is. The subfile names which are searched are of the format: – member-name_CTMN for basic configuration – member-name_CORR for directory configuration – member-name_TRMN, for terminal configuration. *LOAD=name31:* When only the member-name is given, the default binary library name is SYS.DSACORE. The action is equivalent to LOAD=SYS.DSACORE..*name31*.

SV alias **SAVE**

Specifies if the currently created or loaded configurations are to be saved and where. SAVE is effective only if no error has occurred at creation or loading. The configurations currently created in or loaded from a workspace are moved into a binary library. For both SAVE and LOAD, the workspace is specified in ENABLE.
SAVE always involves the whole configuration set being:
basic + directory + terminal.
For incremental generation where ***CRNET=NO*** and ***CRDIR=member-name***, the saved basic and terminal configurations are those stored in the ***permanent*** workspace.
SAVE=NO: ***Default***
No saving action is done.

SAVE=SYSTEM: The generation is saved in the system library SYS.DSACORE into members which are accessible at GCOS 7 initialisation time (RESTART CLEAN). The name of these members are system-reserved names:
- NETGEN_CTMN for basic configuration
- NETGEN_CORR for directory configuration
- NETGEN_TRMN for terminal configuration.

SAVE=file78 of the format: *library-name..member-name*.
It cannot be TEMPRY. The generation is saved in the specified binary library in subfiles whose names are as follows:
- *member-name*_CTMN for network configuration
- *member-name*_CORR for directory configuration
- *member-name*_TRMN for terminal configuration.

SAVE=name31: When only *member-name* is given the default binary library-name is SYS.DSACORE.Then the action is done as for SAVE=SYS.DSACORE..*name31*.

ENBL alias **ENABLE**

Moves the configurations from the ***permanent workspace*** into the ***running workspace*** thereby making them directly accessible to the communications servers.

ENABLE=0: ***Default***
Both the ***running workspace*** and the ***permanent workspace*** are not modified. Where SAVE and LOAD are specified, the ***temporary workspace*** will be used.

ENABLE=1: Enables the configurations from the ***permanent workspace*** in the ***running workspace***.
The ***temporary workspace*** is not concerned.
For creating or loading the basic configuration where *CRNET=member-name*, enabling always involves the whole configuration set being: basic + directory + terminal. All communications servers should be terminated, the session therefore no longer being in progress.
However, for incremental generation where *CRNET=NO* and *CRDIR=member-name*, only the directory configuration is moved between the two workspaces. Enabling can therefore occur while a communications session is in progress.

Notes: Effects of SAVE = SYS and ENABLE = 0 on the members of the binary SYS.DSACORE.

1. For basic generation:
If CRNETGEN is executed with:
"CRNET=name-a SAVE=SYS ENBL=0",

NETGEN_CTMN is modified according to name-a,
NETGEN_CORR becomes empty.

2. For incremental generation:
If CRNETGEN is executed with:
"CRNET=NO CRDIR=name-b SAVE=SYS ENBL=0",

NETGEN_CTMN is saved from the running workspace,
NETGEN_CORR is modified according to name-b.

SRSTFILE

No default. The SRST is used for internal checks each time a basic network configuration is intended to be enabled or saved on a system, other than in TEMP mode. *Directory configuration objects are not concerned in SRST checks.*

- When not specified, the saved or enabled configuration is assumed to be for the DPS 7000 executing the NETGEN. In this case, the SRST used is the local SRST and the local system name supplied in the SYSTEM directive is matched against the DPS 7000 executing the NETGEN.
- When specified, the saved configuration is assumed to be for another DPS 7000. Enabling this configuration is therefore rejected. Checks are made using the specified SRST file and the local system name supplied in the SYSTEM directive is not checked. The SRST file must have been previously prepared using the H_SVSRST utility. Remote SRST always implies *CRNET=member-name* and *ENABLE=0*. The file cannot be TEMPRY.

QFILE

No default. If disk queues are declared in the basic configuration, QFILE is mandatory for supplying the name of the queue file to be formatted. QFILE must always be specified with *CRNET=member-name* even if the enabling is not requested for the current NETGEN run. The file must have been preallocated before starting NETGEN and cannot be deallocated or reallocated without recreating the basic network generation. The actual preformatting of the file will occur at *immediate* or *deferred* enabling of the basic network configuration. The file cannot be TEMPRY.

Knowing which Configuration is Running

It is difficult to know after several successive incremental generations, which configuration is actually running. To solve this difficulty, on normal completion of NETGEN processing involving either LOAD=SYS or SAVE=SYS or ENABLE=1, a source library maintenance step is internally called to move the appropriate configuration descriptions into *remanent* members of SYS.DSACONF. This procedure, however, does not work for private saving or loading. The 4 members concerned are:

- CUR_ENBL_NET which contains the source of the currently enabled basic network configuration
- CUR_ENBL_DIR which contains the source of the currently enabled directory configuration
- CUR_SAVE_NET which contains the source of the basic network configuration most recently saved as SYS
- CUR_SAVE_DIR which contains the source of the directory configuration most recently saved as SYS.

Members CUR_ENBL_* are meaningless after a restart CLEAN but are not destroyed. They are deleted after an enabling associated to LOAD=member-name. Members CUR_SAVE_* correspond to the currently enabled configurations only after an enabling on restart (CLEAN or COLD or WARM) with reply YES or Y. They are deleted after a saving with SAVE=SYSTEM associated to LOAD=member-name.

EXAMPLES:

1. Regeneration from previous GCOS 7 release or First installation for a basic network configuration only:

   ```
   CRNETGEN CRNET=MYNET SAVE=SYSTEM ENABLE;
   ```

2. First installation for basic network and directory configurations:

   ```
   CRNETGEN CRNET=MYNET CRDIR=MYDIR SAVE=SYS ENABLE;
   ```

3. Generating an incremental directory configuration over a basic one already running:

   ```
   CRNETGEN CRDIR=MYDIR SAVE=SYS ENABLE;
   ```

4. Preparing a basic network configuration while a communications session is running:

   ```
   CRNETGEN CRNET=MYNET SAVE=MYBLIB..MYMB;
   ```

5. Enabling the prepared configuration for testing:

   ```
   CRNETGEN LOAD=MYBLIB..MYMB ENABLE;
   ```

6. After test enabling again of the generation which was previously running:

   ```
   CRNETGEN LOAD=SYSTEM ENABLE;
   ```

7. Saving the tested generation as a new system reference:

   ```
   CRNETGEN LOAD=MYBLIB..MYMB SAVE=SYSTEM;
   ```

8. Generating a network and directory configurations for a remote system:

   ```
   CRNETGEN CRNET=RSNG CRDIR=RSDIR SAVE=MYBLIB..RSMB
            SRSTFILE=MYSRST:K448:MS/M452;
   ```

9. Enabling an incremental directory configuration and saving the updated set of configurations:

   ```
   CRNETGEN CRDIR=MYDIR SAVE=SYSTEM ENABLE;
   ```

10. Disabling an already enabled directory configuration:

    ```
    CRNETGEN CRDIR=MYEMPTYDIR ENABLE;
    ```

❑

2.8 OCS

OCS (Open Communication Subsystem) is the communications server associated with OCS Front End to enables a DPS 7000 platform to access to FDDI or high performance Ethernet cable.

Declaration in NETGEN

Each OCS server must be declared in the GCOS 7 NETGEN in order to configure it and to define its related controller.

The related NETGEN directives are:

- SVR directive: describes the OCS server.
- LCT directive: describes the OCS Front End controller.

3. Network Configuration Description (High Level Directives)

3.1 General Rules

Basic Network Configuration

In the source basic network configuration, directives must be entered in the order specified in the chart overleaf between the NET and ENET directives, being the first and last directives. However, once all passthroughs and neighbors have been declared, remote systems may be entered in any order. See Note.

For MCS applications, all remote systems which dialog with QMON must be declared in the Basic Network Configuration.

Directory Configuration

In the source directory configuration, remote systems and the correspondents may be entered in any order between the DIR and EDIR directives, being the first and last directives. However, since their semantics are completely independent, remote systems should be declared together and separated from correspondents. See Note.

Remote systems declared in the Directory Configuration must not include those which dialog with QMON for MCS applications.

NOTE:

The actual NETGEN lexical ordering may influence later processing although it does not affect the access time of the object.

The objects displayed by the DNET operator command are listed either in the exact order of their NETGEN lexical declaration such as TNS objects, or in reverse order for systems, administration and directory objects, and queues.

OPEN 7 ENVIRONMENT

Where the DPS 7000 works in the OPEN 7 environment and communicates using only the TCP/IP protocols, only the SYSTEM directive describing the local system need be supplied.

The following charts give a generalized sequence of a network description.

1. Only the relevant SYSTEM arguments needed for distinguishing the type of the systems are given.
2. The NET directive is mandatory only if queues are declared. Otherwise the NET and ENET directives can be omitted. Low-level COMM directives can appear anywhere.

Basic Configuration

```
[ NETWORK ]

  SYSTEM TYPE=LOC

                  { PF = DN7100,PSI = CChh     }
  SYSTEM TYPE=PT  { PF = CNP7   ,ISL = LAN addr }
   .              { PF = DN7100,ISL = LAN addr }
   .
   .
  SYSTEM TYPE=NB  {             ISL = LAN addr }
   .
   .
   .
  SYSTEM TYPE=RMT
   .
   .
   .
[ Network administration directives ] (see Section 4)
   .
   .
   .
[ MCS queues ]                        (see Appendix D)
   .
   .
   .
[ ENDNETWORK ]
```

Directory Configuration

```
  DIRECTORY

[ SDOM subdomain   { NB = (neighbor-list) } ]

[ SYSTEM TYPE=RMT                                        ]
    .
    .
    .
[ Service of correspondents directives ] (see Sect. 5)
    .
    .
    .
[ ENDDIRECTORY ]
```

1. For all directives, parameters can be specified in any order. All arguments are introduced by their corresponding keyword. NAME is the only exception being the first parameter. It is therefore positional and can be omitted.
2. In any directive when the argument is boolean (0 or 1), the keyword on its own is equivalent to specifying 1 as its argument, for example:

   GENQMON is equivalent to GENQMON=1

   NGENQMON is equivalent to GENQMON=0.

List of Network Configuration High Level Directives

The NETGEN directives described in this Section are arranged in the alphabetical order of their abbreviations. The list below puts the same directives in alphabetical order of their expanded acronyms, giving also the corresponding abbreviation, configuration description destination (B for Basic, D for Directory, Ds for remote systems) and paragraph number.

Abbr	Expanded	Dest	Par.	Abbr	Expanded	Dest	Par.
COMM	COMMENT	BD	3.2	ENET	ENDNETWORK	B	3.6
DEF	DEFAULT	B	3.3	NET	NETWORK	B	3.7
DIR	DIRECTORY	D	3.4	SDOM	SUBDOMAIN	Ds	3.8
EDIR	ENDDIRECTORY	D	3.5	SYS	SYSTEM	BDs	3.9

OCS Environment

OCS is the GCOS 7 telecommunications server associated with the OCS Front End to offer the DPS 7000 platform access to FDDI or high performance Ethernet cable.

The support of OCS Front End requires two levels of configuration:

- at GCOS 7 NETGEN level, to declare the OCS server and its front ends,
- at OCS level, to configure the telecommunications stacks in the OCS Front End .

OCS Front End configuration is described in chapter 8.

3.2 COMM (Comment) Directive

Purpose

The COMM directive can be located anywhere in the network configuration description.

Syntax

```
{ COMM    }
{         } ' string-255 ' ;
{ COMMENT }
```

Parameter Description

string	A comment up to 255 characters and enclosed between simple quotes.

3.3 DEF (Default) Directive

Purpose

The DEFAULT directive applies only to the basic network configuration description. It specifies or modifies some attributes common to several systems. Values set remain effective until changed or reset by a subsequent DEFAULT directive. Using this directive may resolve some tuning or monitoring situations without invoking the overriding mechanism described in the Section 7.

The DEF directive is effective only on systems described through the SYSTEM directive, and will refer to and modify arguments belonging all to the low-level description expanded by each SYSTEM directive. To understand how and when default arguments set by this directive are used, the user must know how the SYSTEM directive works and what it generates in terms of low-level directives. See Section 7.

Syntax Format 1

```
{ DEF     }                                     [          { LIST   } ]
{         }        [, NAME = name4 ]            [, TYPE =  { SET    } ]
{ DEFAULT }                                     [          { RESET  } ]

[           { ANY | ALL        } ]
[, CLASS =  { TS  | TRANSPORT  } ] ;
[           { TP  | PROTOCOL   } ]
[           { CT  | CONTROLLER } ]
```

Syntax Format 2

```
{ DEF     }
{         }         [, NAME = name4 ]            , TYPE = SET
{ DEFAULT }
[   { T1          }    { 10   } ]    [   { T2        }    { 60   } ]
[,  { FLOW        } =  {      } ]    [,  { SILENT    } =  {      } ]
[   { RETRYDELAY  }    { dec2 } ]    [   { IDLEDELAY }    { dec4 } ]

[   { T3    }    { 320  } ]          [   { N        }    { 3    } ]
[,  {       }  = {      } ]          [,  { RETRY    }  = {      } ]
[   { WATCH }    { dec4 } ]          [   { MAXRETRY }    { dec2 } ]

[   { W      }   { 8    } ]          [             { 1 } ]
[,  {        } = {      } ]          [, RMTLOAD =  {   } ]
[   { WINDOW }   { dec2 } ]          [             { 0 } ]

[   { TPDUSZ   }                                                 ]
[,  {          }={ 128 | 256 | 512 | 1024 | 2048 | 4096 | 8192 } ];
[   { TPDUSIZE }                                                 ]
```

Parameter	Description
NAME	*NAME=name4:* Identifies the current DEFAULT directive.
TYPE	Specifies the action to be performed on the default arguments implied by the class: *TYPE=LIST:* ***Default*** Lists the currently active default arguments. *TYPE=RESET:* Erases all current active default arguments of the class so that they no longer affect subsequent generations of SYSTEM directives. *TYPE=SET:* – when associated with CLASS (format 1), the current active default arguments of the class are replaced by their default values as specified in the Format 2 – when associated with any parameters (format 2), the corresponding active default arguments are replaced by those specified in these parameters.
CLASS	**Applicable only to Format 1.** Defines the scope of the action specified by TYPE as follows: CLASS=ANY or ALL: **Default** The action applies to all the classes which follows. *CLASS=TP or PROTOCOL:* The action affects the window size and the TPDU size, being arguments of the transport protocol for the systems subsequently declared as attached to an ISL cable. *CLASS=TS or TRANSPORT:* The action affects the timers, FLOW and SILENT, and the RETRY count, being transport layer arguments for the systems subsequently declared as attached to an ISL cable. *CLASS=CT or CONTROLLER:* either affects the watchdog timer in the local controllers attaching the local system to the ISL cable or inhibits or enables remote loading capability by the local system of a remote controller for systems subsequently declared as attached to the ISL cable. **Note:** CLASS (format 1) is mutually exclusive with all the following parameters which are specific to format 2.

T1 or **FLOW** or **RETRYDELAY**

Applicable only to Format 2. Time-out in seconds before the local transport re-transmits a TPDU when no acknowledgement is received. This default argument **belongs to the TS class** and will be taken into account when subsequent SYSTEM directives describe systems attached to the ISL.

1 <= *flow* <= 30. ***Default:*** 10 seconds.

T2 or **SILENT** or **IDLEDELAY**

Applicable only to Format 2. Time-out in seconds before the connection is released because no TPDU has been received. This protects against unsignalled failures. This default argument **belongs to the TS class** and will be taken into account when subsequent SYSTEM directives describe systems attached to the ISL.

1 <= *silent* <= 600. ***Default:*** 60 seconds.

N or **RETRY** alias **MAXRETRY**

Applicable only to Format 2. Maximum number of times local transport is to re-transmit an un-ACKed TPDU. This default argument **belongs to the TS class** and will be taken into account when the subsequent SYSTEM directives describe systems attached to the ISL.

1 <= *retry* <= 10. ***Default:*** 3.

T3 alias **WATCH**

Applicable only to Format 2. Defines the watchdog delay in tenths of a second between FEPS and the Datanet over the PSI. This value must match that declared for T3 in the CH directive of DNS SYSGEN. This default argument **belongs to the CT class** and will be taken into account when a subsequent SYSTEM directive describes a passthrough linked to the local system via a PSI.
The default value defined by T3 also applies to OCS (WATCH parameter of the LCT directive for OCS).

200 <= *watch* <= 6000. ***Default:*** 320 tenths of a second.
T3=0: no watchdog. Reserved for debugging purpose only.

W alias **WINDOW**

Applicable only to Format 2. Specifies the maximum number of input and output un-ACKed TPDUs for the transport connection. This default argument **belongs to the TP class** and will be taken into account when subsequent SYSTEM directives describe systems attached to the ISL.

1 <= *window* <= 15. ***Default***: 8.

TPDUSZ alias **TPDUSIZE**

Applicable only to Format 2. This default argument **belongs to the TP class** and will be taken into account when subsequent SYSTEM directives describe systems attached to the ISL.

TPDUSZ= 128, 256, 512, 1024, 2048, 4096 or 8192.
Default: 2048.

RMTLOAD

Applicable only to Format 2. Specifies if the remote controller is to accept remote requests for its automatic loading.

RMTLOAD=1: ***Default***, Requests are accepted.
RMTLOAD=0: No remote automatic loading.
This default argument **belongs to the CT class** and will be taken into account when subsequent SYSTEM directives describe systems attached to the ISL.

3.4 DIR (DIRECTORY) Directive

Purpose

This directive applies only to the directory configuration description, it must be unique and be the first not-comment directive of this description.

Syntax

```
{ DIR       }
{           }       [ NAME = name4 ] ;
{ DIRECTORY }
```

Parameter Description

NAME

NAME=name4: When specified, is checked to be the same as the name specified in the NET directive of the basic network configuration description:

either currently running in the case of incremental generation
or created just before the directory itself.

Default: The name in the NET directive.

3.5 EDIR (ENDDIRECTORY) Directive

Purpose

This directive applies only to the directory configuration description. It is optional:

- when present, must be unique and must be placed at the end of the directory configuration description
- when omitted, the end-of-file acts as description boundary for the directory configuration.

Syntax

```
{ EDIR         }
{              }          [ NAME = name4 ] ;
{ ENDDIRECTORY }
```

Parameter Description

NAME *NAME=name4:* Identifies the current directory configuration description.

Optional parameter:

- when specified, must be the name in the DIR directive
- must be omitted, if the name in the DIR directive is omitted.

3.6 ENET (ENDNETWORK) Directive

Purpose

This directive applies only to the basic network configuration description. It is optional:

- when specified, must be the last non-COMM directive to indicate the end of this description, thereby allowing global checks to be done
- when omitted:
  - either the end-of-file when no directory configuration description follows
  - or the DIR directive acts as the boundary of the basic network configuration description.

At the end of the basic network description, if only the local system using TCP/IP protocol is described, a dummy CNP 7 is automatically generated for internal consistency. See SYSTEM and LCT directives.

Syntax

```
{ ENET       }
{            }         [ NAME = name4 ] ;
{ ENDNETWORK }
```

Parameter Description

NAME

NAME=name4: Optional parameter:

- when specified, must be the name in the NET directive
- if the NET directive is omitted, must be the name of the SYSTEM specifying PF=LSYS.

3.7 NET (NETWORK) Directive

Purpose

This directive applies only to the basic network configuration description. It identifies the network generation and provides global information and if needed specific MCS parameters.

It is optional. However if the network generation contains MCS queues, it becomes mandatory as first non-COMM directive before the SYSTEM directive describing the Local System. For the *overriding mechanism*, see Section 7.

Syntax

```
{ NET     }
{         }        NAME = name4
{ NETWORK }

[  { QM      }   { 0 } ]                [  { SIMU    }   { 1    } ]
[, {         } = {   } ]                [, {         } = {      } ]
[  { GENQMON }   { 1 } ]                [  { MAMSIMU }   { dec2 } ]

                                        [                { 20   } ]
[, APPLIB = name-44 ]                   [, RESTARTCNT  = {      } ]
                                        [                { dec3 } ]

[  { PSSW     }          ]              [              { 76   } ]
[, {          } = name10 ]              [, QMBLKSZ =   {      } ]
[  { PASSWORD }          ]              [              { dec4 } ]

[              { 512  } ]               [              { 0    } ]
[, QFBLKSZ =   {      } ]               [, QMPOOL  =   {      } ] ;
[              { 1024 } ]               [              { dec3 } ]
```

Parameter Description

NAME — *NAME=name4:* right-padded with underscores (_) if less than 4 characters. Identifies the current basic network configuration generation. No default.
If the Local System name is omitted in the LSYS directive, its name by default is that of the NET directive.

MCS QUEUE PARAMETERS

All other parameters of this directive are related to MCS queues handling and described in detail in Appendix D.

3.8 SDOM (SUBDOMAIN) Directive

Purpose

SDOM applies only to the directory configuration description. It defines an OSI addressing subdomain in the network that is a set of OSI network addresses which have in common some specified upper digits (mapping NSAP prefix) and which follow the same routing rules.

The Subdomain is an object used for routing between session, transport and network layers. Addresses and location of the passthroughs or neighbors involved in the same subdomain must be consistent.

Syntax (for FEPS, TNS, and OCS)

```
{ SA         }
{ SDOM       }          NAME = name8
{ SUBDOMAIN }

{ { NB | NEIGHBOR } }
{ { NEIGHBOUR     } }              [ { 0    } [ { 7    } ] ]
{,                  } = ( {name4 [:{       } [:{       } ] ] } ...)
{ { PT | PASSTHRU } }              [ { dec1 } [ { dec1 } ] ]
{ { PASS_THROUGH  } }

{  PFX | PREFIX }
{,              } = hexa40 ;
{  NSAP_PREFIX  }
```

Parameter Description

NAME *NAME=name8:* Mandatory, identifies the Subdomain. No default.

PFX alias **PREFIX** alias **NSAP_PREFIX**
PFX=hexa40: Specifies the prefix for OSI Network Service Access Point addresses typical of the current subdomain.

NSAP ALLOCATION

Allocating NSAP addresses to OSI applications is the responsibility of the system or network administrators and must conform to the standards defined by the network addressing authorities such as ISO and CCITT.

This allocation must be hierarchical in order to enable:

- reducing the number of Subdomain objects in the configuration
- adding new OSI remote systems without updating the configuration.

SUBDOMAIN SELECTION

The session routing function searches among the subdomain objects those with an NSAP_PREFIX matching the leftmost digits of the NSAP address of the OSI application requesting connection.

If no prefix matches the NSAP, no subdomain is selected and all the configured routes can be successively tried to access the remote system.

If one or more prefixes match the NSAP, the longest matching prefix wins. The NEIGHBOR parameter of the selected subdomain gives the list of routes which can be used for the connection.

NB (NEIGHBOR alias **NEIGHBOUR)** ***and***
PT (PASSTHRU alias **PASS_THROUGH)**

Lists up to 32 neighbors or passthroughs providing possible routes associated with this subdomain.

Each object can be:

- an OCS Front End,
- a neighbor system,
- a passthrough system.

For the OCS Front End, the object name is the OCS Front End name given in the LCT directive.
Each listed object may specify its own range of class of service by a pair of single decimal digits preceded by a semi-colon. The left value is the minimum class, the right one is the maximum class. Default maximum class is the minimum class.

0<=min<=max<=7.

Default is 0:7 meaning that all the classes of service are assumed as defined.

NB or PT=(name4 [:dec1 [:dec1]] ...), where

name4 is the name of the system

:dec1 [:dec1] is the range of *class of service*, the left value being the minimum class, the right one the maximum.

0 <= min <= max <= 7. Default maximum is the minimum.

***Default**:* 0:7, includes all the classes of service.

SUBDOMAIN ROUTE SELECTION

QOS (Quality Of Service) and subnetwork selection dynamically choose the route among those offered by the selected subdomain. QOS determines priorities, while subnetwork selection handles the classes of service statically attached to the routes.

QOS

If no *class of service* is defined for the selected subdomain routes, all the routes have the same priority with the default range 0,7.

If *class of service* is defined for some routes of the selected subdomain, the routes may be weighted differently depending on their ranges.

Each OSI application may require a **desired** and a **minimum** *class of service* to execute. If the OSI application does not request a *class of service*, any *class of service* meaning **minimum** is assumed.

Subnetwork Selection

Subnetwork selection searches from left to right, the list of routes of the selected subdomain.

Routes whose range of *class of service* matches the **desired** *class of service* of the application, are selected and successively tried for connection.

If all routes thus selected are unsuccessful or if none is selected, those matching the **minimum** are searched for and tried.

If all routes are unsuccessful or none is selected, the connection is rejected.

3.9 SYS (SYSTEM) Directive for FEPS, TNS, and OCS

Purpose

The SYSTEM directive applies to:

- **all syntax formats of the basic network configuration description**
- **and to syntax format 4 only of the directory configuration description.**

Each SYSTEM directive enables the user to fully declare a system in a single directive.

Sections 6 and 7 treat in detail how low-level communications objects relating to the system are generated and named.

NOTE:

In the syntax for PROFILE, *if several releases apply for a system,* the release taken as default when its abbreviated profile is specified, is shown in ***bold italics***.

The allowed PROFILE depends on the DPS7000 system model when the TYPE is Passthrough or Neighbor.

MIXING LOW-LEVEL DIRECTIVES

Low-level directives should only be used to override some communications attributes.

It is highly recommended that at least the Local System and all passthroughs are described by SYSTEM directives.

See Section 7 for details on mixing high and low-level directives. See Section 6 for the low-level description of the network.

3.9.1 OSI System Description

OSI Syntax Format 1: Local System

```
{ SYS     }
{         }          NAME = name4         [,TYPE = { LOC | LOCAL }]
{ SYSTEM }

[ { PF       }    { LSYS                                   } ]
[,{          } = {                                          } ]
[ { PROFILE }    { DPS7 | 'DPS7/GCOS-7/{V9|V10}'           } ]

[ { PID      }    { ISO    } ]             { TSEL        }
[,{ PIDTYPE } = {          } ]            ,{             } = hexa64
[ { PIDTSEL }    { hexa64 } ]             { TSELECTOR   }

[,NSAP = ( hexa40 [ hexa40 ]...)]

,SCID = dec3 : dec3

[    { ISL  }      {G}                                  {CBL_MAIN}     ]
[   ,{      } = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [,{          }])   ]
[    { ISL1 }      {L}                                  { name8   }     ]
[                                                                        ]
[ [               {G}                                              ] ]
[ [     ,ISL2 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])    ] ]
[ [               {L}                                              ] ]
[ [                                                                ] ]
[ [ [             {G}                                            ] ] ]
[ [ [   ,ISL3 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])  ] ] ]
[ [ [             {L}                                            ] ] ]
[ [ [                                                            ] ] ]
[ [ [ [           {G}                                          ] ] ] ]
[ [ [ [,ISL4 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])] ] ] ]
[ [ [ [           {L}                                          ] ] ] ]

[ { POOLSZ   }        { 1    }     { 4    }    ]
[,{          } = ( [ {       }, ] {        } ) ]
[ { POOLSIZE }        { dec3 }     { dec3 }    ]

[ { TCP   }    { 0 } ]
[,{       } = {     } ]
[ { TCPIP }    { 1 } ]

[ { MADDR }      {G}                                                     ]
[,{       } = {[{ }:] xx-xx-xx-xx-xx-xx, xx-xx-xx-xx-xx-xx [,...]}]
[ { MADDR1}      {L}                                                     ]
```

```
[[               {G}                                                  ]]
[[  , MADDR2 = {[{ }:] xx-xx-xx-xx-xx-xx, xx-xx-xx-xx-xx-xx [,...]}   ]]
[[               {L}                                                  ]]
[[                                                                    ]]
[[[              {G}                                                 ]]]
[[[ , MADDR3 = {[{ }:] xx-xx-xx-xx-xx-xx, xx-xx-xx-xx-xx-xx [,...]}  ]]]
[[[              {L}                                                 ]]]
[[[                                                                  ]]]
[[[[             {G}                                                ]]]]
[[[[, MADDR4 = {[{ }:] xx-xx-xx-xx-xx-xx, xx-xx-xx-xx-xx-xx [,...]} ]]]]
[[[[             {L}                                                ]]]]

[ { RADDR  }   {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF          }       ]
[,{        } = {                                                 }       ]
[ { RADDR1 }   {  ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx [,...]) }       ]
[                                                                        ]
[[             {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF          }      ]]
[[  , RADDR2 = {                                                 }      ]]
[[             {  ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx [,...]) }      ]]
[[                                                                      ]]
[[[            {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF          }     ]]]
[[[ , RADDR3 = {                                                 }     ]]]
[[[            {  ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx [,...]) }     ]]]
[[[                                                                    ]]]
[[[[           {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF          }    ]]]]
[[[[, RADDR4 = {                                                 }    ]]]]
[[[[           {  ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx [,...]) }    ]]]]

[  { NCC  }      {G}                                       ]
[ ,{      } = ([{ }:] XX-XX-XX-XX-XX-XX, EAhexa2 }        ]
[  { NCC1 }      {L}                                       ]
[                                                          ]
[[               {G}                                      ]]
[[    , NCC2 = ([{ }:] XX-XX-XX-XX-XX-XX, EAhexa2 }       ]]
[[               {L}                                      ]]
[[                                                        ]]
[[[              {G}                                     ]]]
[[[   , NCC3 = ([{ }:] XX-XX-XX-XX-XX-XX, EAhexa2 }      ]]]
[[[              {L}                                     ]]]
[[[                                                      ]]]
[[[[             {G}                                    ]]]]
[[[[  , NCC4 = ([{ }:] XX-XX-XX-XX-XX-XX, EAhexa2 }     ]]]]
[[[[             {L}                                    ]]]]

[ { OBJLIST     }    { 0 } ]
[,{             } =  {   } ]
[ { OBJECT_LIST }    { 1 } ]

[ { OBJPFX        }    { 1 } ]
[,{               } =  {   } ] ;
[ { OBJECT_PREFIX }    { 0 } ]
```

OSI Syntax Format 2: Passthrough

OSI Syntax Format 2-1 : Passthrough for DPS 7000:

```
{ SYS    }
{        }          NAME = name4
{ SYSTEM }

[ ,TYPE = { PT | PASSTHRU | PASS_THROUGH } ]

[  { PID     }   { ISO    } ]
[, { PIDTYPE } = {        } ]
[  { PIDTSEL }   { hexa64 } ]

, NSAP = ( hexa40 [ hexa40 ]... )

[  { PF      }   { DN7100 | 'DN7100/DN-C/{V3.0|V4.1}' } ]
[, {         } = {                                    } ]
[  { PROFILE }   { CNP7   | 'CNP7/CNS7/{A0|A1|A2'     } ]

,SCID = dec3 : dec3

[,PSI = ( CChexa2 [ ,CChexa2 ] )                 ]
[                                                ]
[ [ { POOLSZ   }      { 1    }   { 4    }    ] ]
[ [,{          } = ( [ {      },] {      } ) ] ]
[ [ { POOLSIZE }      { dec2 }   { dec2 }    ] ]

[   { ISL  }      {G}                        { CBL_MAIN }      ]
[  ,{      } = ( [ { }: ] hh-hh-hh-hh-hh-hh, [,{          }] )  ]
[   { ISL1 }      {L}                        { name8    }      ]
[                                                              ]
[ [               {G}                                        ] ]
[ [    ,ISL2 = ( [ { }: ] hh-hh-hh-hh-hh-hh,  [,name8] )      ] ]
[ [               {L}                                        ] ]
[ [                                                          ] ]
[ [ [             {G}                                      ] ] ]
[ [ [  ,ISL3 = ( [ { }: ] hh-hh-hh-hh-hh-hh,  [,name8] )    ] ] ]
[ [ [             {L}                                      ] ] ]
[ [ [                                                      ] ] ]
[ [ [ [           {G}                                    ] ] ] ]
[ [ [ [,ISL4 = ( [ { }: ] hh-hh-hh-hh-hh-hh,  [,name8] )  ] ] ] ]
[ [ [ [           {L}                                    ] ] ] ]

[        { 1 } ]
[,LOAD = {   } ]
[        { 0 } ]

[ { OBJLIST     }   { 0 } ]
[,{             } = {   } ] ;
[ { OBJECT_LIST }   { 1 } ]
```

OSI Syntax Format 2-2 : Passthrough for DPS 7000/XTA:

```
{ SYS    }
{        }           NAME = name4
{ SYSTEM }

[ ,TYPE = { PT | PASSTHRU | PASS_THROUGH } ]

[  { PID }       { ISO    } ]
[, { PIDTYPE } = {        } ]
[  { PIDTSEL }   { hexa64 } ]

, NSAP = ( hexa40 [ hexa40 ]... )

  { PF      }
, {         } =  { MW2600LE | 'MW2600LE/DNS-E/V3' }
  { PROFILE}

, SCID = dec3 : dec3

  { ISL  }       {G}                          { CBL_MAIN }
, {      } = ( [ { }: ] hh-hh-hh-hh-hh-hh [,{          } ] )
  { ISL1 }       {L}                          { name8    }

[        { 1 } ]
[,LOAD = {   } ]
[        { 0 } ]

[ { OBJLIST     }   { 0 } ]
[,{             } = {   } ] ;
[ { OBJECT_LIST }   { 1 } ]
```

OSI Syntax Format 3: Neighbor

OSI Syntax Format 3-1 : Neighbor for DPS 7000:

```
{ SYS    }
{        }           NAME = name4
{ SYSTEM }

[, TYPE = { NB | NEIGHBOR | NEIGHBOUR } ]

[  { PID     }   { ISO    } ]
[, { PIDTYPE } = {        } ]
[  { PIDTSEL }   { hexa64 } ]

, NSAP = ( hexa40 [ hexa40 ]... )

[                { DN7100 | 'DN7100/DN-C/{V3.0|V4.1}'         } ]
[                { CNP7   | 'CNP7/CNS7/{A0|A1|A2}'            } ]
[  {PF      }    { DPS7   | 'DPS7/GCOS-7/{V9|V10}'            } ]
[, {        } =  { STID   | 'STID/ISO/{SID3|SID4}'            } ]
[  {PROFILE }    { NIP                                        } ]
[                { DPS6   | 'DPS6/DSS/{2.2|3.0|3.1|4.0|4.1}'  } ]
[                { DPS6P  | 'DPS6P/HVS/{1.0|2.0}'             } ]
[                { MW2600 | 'MW2600/DNS-E/V3'                 } ]

 {ISL }      {G}                                    {CBL_MAIN}
,{    }  = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, {        }])
 {ISL1}      {L}                                    { name8  }

[            {G}                                          ]
[  ,ISL2 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])  ]
[            {L}                                          ]
[                                                         ]
[[           {G}                                         ]]
[[ ,ISL3 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8]) ]]
[[           {L}                                         ]]
[[                                                       ]]
[[[          {G}                                        ]]]
[[[,ISL4 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8]) ]]]
[[[          {L}                                        ]]]

[        { 0 } ]
[,LOAD = {   } ]
[        { 1 } ]

[ { OBJLIST     }   { 0 } ]
[,{             } = {   } ] ;
[ { OBJECT_LIST }   { 1 } ]
```

OSI Syntax Format 3-2 : Neighbor for DPS 7000/XTA:

```
{ SYS    }
{        }          NAME = name4
{ SYSTEM }

[ ,TYPE = { NB | NEIGHBOR | NEIGHBOUR } ]

[  { PID }        { ISO    } ]
[, { PIDTYPE } = {        } ]
[  { PIDTSEL }    { hexa64 } ]

, NSAP = ( hexa40 [ hexa40 ]... )

  { PF      }
, {         } =  { MW2600LE | 'MW2600LE/DNS-E/V3' }
  { PROFILE}

  { ISL  }       {G}                          { CBL_MAIN }
, {      } = ( [ { }: ] hh-hh-hh-hh-hh-hh [,{          } ] )
  { ISL1 }       {L}                          { name8    }

[        { 1 } ]
[,LOAD = {   } ]
[        { 0 } ]

[ { OBJLIST     }   { 0 } ]
[,{             } = {   } ] ;
[ { OBJECT_LIST }   { 1 } ]
```

OSI Syntax Format 4: Remote System

```
{ SYS    }
{        }          NAME = name4
{ SYSTEM }

,TYPE = { RMT | REMOTE }

[   { PID     }   { ISO    } ]
[,  { PIDTYPE } = {        } ]
[   { PIDTSEL }   { hexa64 } ]

, NSAP = ( hexa40 [ hexa40 ]... )

                { DPS7     |'DPS7/GCOS-7/{V9|V10}'                }
                { DPS6     |'DPS6/DSS/{2.2|3.0|3.1|4.0|4.1}'      }
                { DPS6P    |'DPS6P/HVS/{ 1.0 | 2.0 }'             }
                { DN7100   |'DN7100/DN-C/{ V3.0|V4.1}'            }
 { PF       }   {           'DN7100/DN-B/{ V2.5 | V2.6 }'         }
,{          } = { DPS8     |'DPS8/GCOS-8/SR{2500|3000|4000|5000}' }
 { PROFILE  }   { DPS4     |'DPS4/GCOS-4/{ NCT1 | NCT2 }'         }
                { CNP7     |'CNP7/CNS7/{ A0 | A1 | A2}'           }
                { M400     |'M400/DSA/{ 3.0 | 3.1 | 4.0 }'        }
                { STID     |'STID/ISO/{SID3|SID4}'                }
                { IBM      |'IBM/JANUS/DN300'                     }
                { MW2600   |'MW2600/DNS-E/V3'                     }
[               { MW2600LE|'MW2600LE/DNS-E/V3'                    }]

[ { OBJLIST     }   { 0 } ]
[,{             } = {   } ] ;
[ { OBJECT_LIST }   { 1 } ]
```

3.9.2 DSA System Description

DSA Syntax Format 1: Local System

```
{ SYS    }
{        }          NAME = name4
{ SYSTEM }

[ , TYPE = { LOC | LOCAL } ]

[ { PF      }   { LSYS                              } ]
[,{         } = {                                   } ]
[ { PROFILE }   { DPS7 | 'DPS7/GCOS-7/{V9|V10}'    } ]

,SCID = dec3 : dec3

[  { PID         }   { DSA } ]
[, {             } = {     } ]
[  { PIDTYPE     }   { SID } ]

[  { ISL  }     {G}                                    {CBL_MAIN}  ]
[ ,{      } = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [,{         }]))]
[  { ISL1 }     {L}                                    { name8   }  ]
[[                                                                ]]
[[            {G}                                                 ]]
[[  ,ISL2 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])         ]]
[[            {L}                                                 ]]
[[                                                                ]]
[[[           {G}                                                ]]]
[[[ ,ISL3 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])        ]]]
[[[           {L}                                                ]]]
[[[                                                              ]]]
[[[[          {G}                                               ]]]]
[[[[,ISL4 = ([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])       ]]]]
[[[[          {L}                                               ]]]]

[ { POOLSZ   }       { 1    }    { 4    }    ]
[,{          } = ( [ {      }, ] {      } ) ]
[ { POOLSIZE }       { dec3 }    { dec3 }    ]

[ { TCP   }   { 0 } ]
[,{       } = {   } ]
[ { TCPIP }   { 1 } ]
```

```
[ { MADDR }      {G}                                                    ]
[,{       } = {[{ }:]xx-xx-xx-xx-xx-xx, xx-xx-xx-xx-xx-xx [,...]}       ]
[ { MADDR1}      {L}                                                    ]
[                                                                       ]
[[               {G}                                                   ]]
[[  ,MADDR2 = {[{ }:] xx-xx-xx-xx-xx-xx,xx-xx-xx-xx-xx-xx [,...]}      ]]
[[               {L}                                                   ]]
[[                                                                     ]]
[[[              {G}                                                  ]]]
[[[ ,MADDR3 = {[{ }:] xx-xx-xx-xx-xx-xx, xx-xx-xx-xx-xx-xx [,...]} ]]]
[[[              {L}                                                  ]]]
[[[                                                                   ]]]
[[[[             {G}                                                 ]]]]
[[[[,MADDR4 = {[{ }:] xx-xx-xx-xx-xx-xx, xx-xx-xx-xx-xx-xx [,...]}]]]]
[[[[             {L}                                                 ]]]]
[[[[                                                                 ]]]]

[ { RADDR  }   {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF         }      ]
[,{        } = {                                               }      ]
[ { RADDR1 }   {  ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx [,...]) }     ]
[                                                                     ]
[[             {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF         }     ]]
[[   ,RADDR2 = {                                               }     ]]
[[             {  ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx [,...]) }    ]]
[[                                                                   ]]
[[[            {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF         }    ]]]
[[[  ,RADDR3 = {                                               }    ]]]
[[[            {  ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx [,...]) }   ]]]
[[[                                                                 ]]]
[[[[           {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF         }   ]]]]
[[[[ ,RADDR4 = {                                               }   ]]]]
[[[[           {  ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx [,...]) }  ]]]]
[[[[                                                               ]]]]

[  { NCC  }      {G}                                    ]
[, {      }  = ([{ }:] XX-XX-XX-XX-XX-XX, EAhexa2 }     ]
[  { NCC1 }      {L}                                    ]
[                                                       ]
[[               {G}                                   ]]
[[    ,NCC2  = ([{ }:] XX-XX-XX-XX-XX-XX, EAhexa2 }    ]]
[[               {L}                                   ]]
[[                                                     ]]
[[[              {G}                                  ]]]
[[[   ,NCC3  = ([{ }:] XX-XX-XX-XX-XX-XX, EAhexa2 }   ]]]
[[[              {L}                                  ]]]
[[[                                                   ]]]
[[[[             {G}                                 ]]]]
[[[[  ,NCC4  = ([{ }:] XX-XX-XX-XX-XX-XX, EAhexa2 } ]]]]
[[[[             {L}                                 ]]]]
```

```
[ { PT            }                              ]
[,{ PASSTHRU      } = ( name4  [ ,name4  [,...] ] ) ]
[ { PASS_THROUGH }                              ]

[ { OBJLIST     }    { 0 } ]
[,{             } = {     } ]
[ { OBJECT_LIST }    { 1 } ]

[ { OBJPFX        }    { 1 } ]
[,{               } = {     } ]  ;
[ { OBJECT_PREFIX }    { 0 } ]
```

DSA Syntax Format 2: Passthrough

DSA Syntax Format 2-1 : Passthrough for DPS 7000:

```
{ SYS    }
{        }              NAME = name4
{ SYSTEM }

[ ,TYPE = { PT | PASSTHRU | PASS_THROUGH } ]

[  {PF       }   { DN7100 | 'DN7100/DN-C/{V3.0|V4.1}' } ]
[ ,{         } = {                                      } ]
[  {PROFILE }   { CNP7   | 'CNP7/CNS7/{A0|A1|A2'        } ]

,SCID = dec3  : dec3

[   { PID        }    { DSA } ]
[,  {            } = {     } ]
[   { PIDTYPE    }    { SID } ]

[,PSI = ( CChexa2 [ ,CChexa2 ] )                   ]
[                                                  ]
[ [ { POOLSZ   }        { 1    }    { 4    }    ] ]
[ [,{          } = ( [ {       },] {       } ) ] ]
[ [ { POOLSIZE }        { dec2 }    { dec2 }    ] ]

[    { ISL  }       {G}                              { CBL_MAIN }    ]
[   ,{      } = ( [ { }: ]  hh-hh-hh-hh-hh-hh,   [,{            }] )]
[    { ISL1 }       {L}                              { name8    }    ]
[                                                                    ]
[ [                 {G}                                          ] ]
[ [     ,ISL2 = ( [ { }: ]   hh-hh-hh-hh-hh-hh,   [,name8] )     ] ]
[ [                 {L}                                          ] ]
[ [                                                              ] ]
[ [ [               {G}                                        ] ] ]
[ [ [   ,ISL3 = ( [ { }: ]   hh-hh-hh-hh-hh-hh,   [,name8] )   ] ] ]
[ [ [               {L}                                        ] ] ]
```

```
[ [ [                                                         ] ] ]
[ [ [ [           {G}                                       ] ] ] ]
[ [ [ [,ISL4 = ( [ { }: ]   hh-hh-hh-hh-hh-hh,   [,name8] )] ] ] ]
[ [ [ [           {L}                                       ] ] ] ]

[        { 1 } ]
[,LOAD = {   } ]
[        { 0 } ]


[ { PT           }                              ]
[,{ PASSTHRU     } = ( name4 [ ,name4 [,...] ] ) ]
[ { PASS_THROUGH }                              ]

[ { OBJLIST     }   { 0 } ]
[,{             } = {   } ] ;
[ { OBJECT_LIST }   { 1 } ]
```

DSA Syntax Format 2-2 : Passthrough for DPS 7000/XTA:

```
{ SYS    }
{        }                        NAME = name4
{ SYSTEM }

[ ,TYPE = { PT | PASSTHRU | PASS_THROUGH } ]

  { PF      }
, {         } =  { MW2600LE | 'MW2600LE/DNS-E/V3' }
  { PROFILE}

, SCID = dec3 : dec3

[  { PID }        { DSA } ]
[, { PIDTYPE } = {     } ]
[  { PIDTSEL }    { SID } ]

  { ISL  }        {G}                          { CBL_MAIN }
, {      } = ( [ { }: ] hh-hh-hh-hh-hh-hh [,{          } ] )
  { ISL1 }        {L}                          { name8    }

[         { 1 } ]
[,LOAD = {   } ]
[         { 0 } ]

{ PT            }
{,PASSTHRU      } = ( name4 [ ,name4 [,...] ] )
{ PASS_THROUGH }

[ { OBJLIST     }    { 0 } ]
[,{             } = {   } ] ;
[ { OBJECT_LIST }    { 1 } ]
```

DSA Syntax Format 3: Neighbor

DSA Syntax Format 3-1: Neighbor for DPS 7000:

```
{ SYS    }
{        }            NAME = name4
{ SYSTEM }

[, TYPE = { NB | NEIGHBOR | NEIGHBOUR } ]

[                { DN7100 | 'DN7100/DN-C/{V3.0|V4.1}'          } ]
[                { CNP7   | 'CNP7/CNS7/{A0|A1|A2}'             } ]
[  {PROFILE }    { DPS7   | 'DPS7/GCOS-7/{V9|V10}'             } ]
[, {        } =  { STID   | 'STID/ISO/{SID3|SID4}'             } ]
[  {PF      }    { NIP                                         } ]
[                { DPS6   | 'DPS6/DSS/{2.2|3.0|3.1|4.0|4.1}'  } ]
[                { DPS6P  | 'DPS6P/HVS/{1.0|2.0}'              } ]
[                { MW2600 | 'MW2600 /DNS-E/V3'                 } ]

,SCID = dec3 : dec3

[  { PID        }   { DSA } ]
[, {            } = {     } ]
[  { PIDTYPE    }   { SID } ]

 {ISL }     {G}                                     {CBL_MAIN}
,{    } =([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, {        }])
 {ISL1}     {L}                                     {name8   }
[           {G}                                              ]
[   ,ISL2=([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])      ]
[           {L}                                              ]
[[          {G}                                             ]]
[[ ,ISL3=([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])      ]]
[[          {L}                                             ]]
[[[         {G}                                            ]]]
[[[,ISL4=([{ }:] hh-hh-hh-hh-hh-hh, EAhexa2 [, name8])     ]]]
[[[         {L}                                            ]]]

[         { 0 } ]
[,LOAD =  {   } ]
[         { 1 } ]

[ { PT           }                                ]
[,{ PASSTHRU     } = ( name4  [ ,name4  [,...] ] ) ]
[ { PASS_THROUGH }                                ]

[ { OBJLIST     }    { 0 } ]
[,{             }  = {   } ] ;
[ { OBJECT_LIST }    { 1 } ]
```

DSA Syntax Format 3-2 : Neighbor for DPS 7000/XTA:

```
{ SYS    }
{        }           NAME = name4
{ SYSTEM }

[ ,TYPE = { NB | NEIGHBOR | NEIGHBOUR } ]

  { PF      }
, {         } =  { MW2600LE | 'MW2600LE/DNS-E/V3'}
  { PROFILE}

, SCID = dec3 : dec3

[  { PID }        { DSA } ]
[, { PIDTYPE } = {       } ]
[  { PIDTSEL }    { SID } ]

 { ISL  }        {G}                          { CBL_MAIN }
,{      } = ( [ { }: ] hh-hh-hh-hh-hh-hh [,{           } ] )
 { ISL1 }        {L}                          { name8    }

[         { 0 } ]
[,LOAD = {     } ]
[         { 1 } ]

{ PT             }
{,PASSTHRU       } = ( name4 [ ,name4 [,...] ] )
{ PASS_THROUGH }

[ { OBJLIST     }    { 0 } ]
[,{             } = {     } ] ;
[ { OBJECT_LIST }    { 1 } ]
```

DSA Syntax Format 4: Remote System

```
{ SYS    }
{        }          NAME = name4
{ SYSTEM }

, TYPE = { RMT | REMOTE }

               { DPS7     | 'DPS7/GCOS-7/{V9|V10}'                  }
               { DPS6     | 'DPS6/DSS/{2.2|3.0|3.1|4.0|4.1}'        }
               { DPS6P    | 'DPS6P/HVS/{ 1.0 | 2.0 }'               }
               { DN7100   | 'DN7100/DN-C/{ V3.0|V4.1}'              }
 { PF      }   {            'DN7100/DN-B/{ V2.5 | V2.6 }'           }
,{         } = { DPS8     | 'DPS8/GCOS-8/SR{2500|3000|4000|5000}'   }
 { PROFILE }   { DPS4     | 'DPS4/GCOS-4/{ NCT1 | NCT2 }'           }
               { CNP7     | 'CNP7/CNS7/{A0|A1|A2}'                  }
               { M400     | 'M400/DSA/{3.0|3.1|4.0 }'               }
               { STID     | 'STID/ISO/{SID3|SID4}'                  }
               { IBM      | 'IBM/JANUS/DN300'                       }
               { MW2600   | 'MW2600/DNS-E/V3'                       }
[              { MW2600LE| 'MW2600LE/DNS-E/V3'                      }]

,SCID = dec3 : dec3

[  { PID        }   { DSA } ]
[, {            } = {     } ]
[  { PIDTYPE    }   { SID } ]

{ PT            }
{,PASSTHRU      } = ( name4 [ ,name4 [,...] ] )
{ PASS_THROUGH  }

[ { OBJLIST     }   { 0 } ]
[,{             } = {   } ] ;
[ { OBJECT_LIST }   { 1 } ]
```

Parameter Description

NAME

NAME=name4: Identifies the system and is the only mandatory keyword that can be specified as a positional. When PF=LSYS and no NETWORK directive is specified, this name becomes that of the network generation.

NAME=BCLS: **Applies only to Remote Systems (Format 4).** In the basic network configuration description, when the Local System is declared *secured* (PF=LSYS and PT parameters are specified), the system-reserved name BCLS represents an internally generated dummy remote system used by the network security control service.

TYPE

When omitted, the resulting default type is set according to the syntax format recognized through other SYSTEM parameters and by the profile which then becomes mandatory:

SYSTEM parameters (TYPE not specified)			resulting system TYPE
PROFILE	PSI	ISL1	
LSYS	error	optional	LOCAL
DN7100	present	optional	PASSTHROUGH
DN7100	missing	present	PASSTHROUGH
CNP7	error	present	PASSTHROUGH
DPS 7000	error	present	NEIGHBOR
STID	error	present	NEIGHBOR
NIP	error	present	NEIGHBOR
DPS6	error	present	NEIGHBOR
DPS6P	error	present	NEIGHBOR
other	error	error	REMOTE
any	missing	missing	REMOTE

TYPE=LOC or LOCAL: **Applies only to the basic network configuration.** Defines the current system as LOCAL which must be the *first* SYSTEM directive.

The chart below shows the compatibility of type LOCAL with some other SYSTEM parameters:

Local SYSTEM parameters			resulting system TYPE
PROFILE	PSI	ISL1	
LSYS	missing	optional	LOCAL
DPS 7000	missing	optional	LOCAL
other	missing	optional	error
LSYS	present	optional	error
DPS 7000	present	optional	error

TYPE=PT or PASSTHRU or PASS_THROUGH:

Applies only to the basic network configuration and can be listed in the PT parameter of a subsequent SYSTEM directive or in the NB or PT parameter of an SDOM directive.

Defines the current system as PASSTHROUGH which must appear before REMOTE SYSTEMs and, with DSA addressing, must appear before any SYSTEM specifying its name in the PT parameter.

The chart below shows the compatibility of type PASSTHROUGH with some other SYSTEM parameters:

Passthrough SYSTEM parameters			resulting system TYPE
PROFILE	PSI	ISL1	
DN7100	optional	optional	PASSTHROUGH
CNP7	optional	present	PASSTHROUGH
other	optional	optional	error
DN7100	missing	missing	error
CNP7	missing	missing	error
CNP7	optional	missing	error

For DPS 7000/XTA: The MW2600LE is the only allowed PROFILE.

TYPE=NB or NEIGHBOR or NEIGHBOUR:

Applies only to the basic network configuration and can be listed in the NB or PT parameter of an SDOM directive, but not in the PT parameter of a subsequent SYSTEM directive.
Defines the current system as NEIGHBOR which must appear before REMOTE SYSTEMs.
Specifying NEIGHBOR is mandatory if a Datanet or CNP 7 attached on the ISL is not to be used as a passthrough by the Local System.

The chart below shows the compatibility of type NEIGHBOR with some other SYSTEM parameters:

SYSTEM parameters (TYPE=NEIGHBOR)			resulting system TYPE
PROFILE	PSI	ISL1	
DN7100	missing	present	NEIGHBOR
CNP7	missing	present	NEIGHBOR
DPS 7000	missing	present	NEIGHBOR
STID	missing	present	NEIGHBOR
NIP	missing	present	NEIGHBOR
DPS6	missing	present	NEIGHBOR
DPS6P	missing	present	NEIGHBOR
other	missing	present	error
any	missing	missing	error
any	present	present	error

For DPS 7000/XTA : The MW2600LE is the only allowed PROFILE.

TYPE=RMT or REMOTE:

Applies to the basic network and directory configurations.

Defines the current system as REMOTE which in the basic network configuration, must appear after all passthroughs and neighbors.
The chart below shows the compatibility of type LOCAL with other SYSTEM parameters:

Remote SYSTEM parameters			resulting system TYPE
PROFILE	PSI	ISL1	
LSYS	missing	missing	error
NIP	missing	missing	error
other	missing	missing	REMOTE
any	missing	present	error
any	present	missing	error

MW2600/DNS-E/V3 is applicable only to

DPS 7000/XTA.

PF alias **PROFILE**

Defines the profile of the system and is mandatory when TYPE is omitted. When TYPE is specified and PROFILE is omitted, the resulting default is the brief profile shown underlined in the Syntax Format.

PF=LSYS: When TYPE=LOC is omitted, defines the current system as LOCAL corresponding to DPS7. Must be specified only in the first SYSTEM directive.

PF=name6: Brief profile
The system capabilities assumed are those of the most recent hardware and software for this system. Single quotes are optional for this form of profile.

PF=char22: Extended profile
Identifies the hardware, software and release of the system, enclosed between single quotes. For neighbors, the profiles listed are those corresponding to systems which can be currently be attached over the same ISL as the Local System.

MW2600/DNS-E/V3 is applicable to Neighbor or Remote systems.

DPS 7000/XTA: **MW2600LE/DNS-E/V3** is applicable to Remote systems. It is applicable to Passthrough or Neighbor systems only for administration MainWay 2600LE.

DPS 7000 FILE TRANSFER CAPABILITIES

When the remote system is a DPS 7000, its file transfer capability towards the local DPS 7000 is security for the file transfer depending on how the acceptor DPS 7000 system (local or remote) is configured with respect to the file transfer application.

- if locally configured as **protected**, the file will be transferred only after the acceptor agrees to the submitter's access rights.
- if locally configured as **not protected**, there are no restrictions on the file transfer. See 1.

TSEL alias **TSELECTOR**

Applicable only to OSI Format 1 (and mandatory). Defines the default transport layer selector in the OSI addressing of the Local System and is used only when the address is not supplied by the application.

TSEL=hexa-64. No default.

NSAP

Applicable only to and mandatory for OSI Formats. Lists up to 20 Network Service Access Points.

NSAP=(hexa40,hexa40...). No default.

NSAP OF TNS

For OSI Format 1, when TNS is used, the first NSAP in the list is, by internal convention, that associated with TNS.

LOCAL TSEL AND NSAP PARAMETERS

For compatibility with previous GCOS 7 releases supporting full ISO session layer services, NSAP and TSEL parameters specified in DSA Format 1 are not flagged as errors.

PID alias **PIDTYPE** alias **PIDTSEL**

Determines the addressing to be used depending on the architecture.

When applied to OSI Formats:

PID=ISO: Specifies a default transport layer selector for the OSI addressing of the local or remote system PID.

PID=hexa64: Specifies the transport layer selector for the OSI addressing of the local or remote system PID.

When applied to DSA Formats:

PID=SID: Specifies that the addressing is DSA and the protocol is SID for which a PID protocol is used.

PID=DSA: ***Default****,* specifies that the addressing is DSA and the protocol is DSA200. ***No PID is activated.***

SCID	**Mandatory for OSI Formats 1 and 2, and all DSA Formats.** Defines the Session Control Identifier being the network-wide and unique DSA address of the system in the network. The administration of the local front-end processor by the FECM server is based on DSA addressing, whether in the DSA or ISO environment. SCID therefore continues to be used in the description of the Local System and its passthroughs. *SCID=dec3:dec3*, where 0 < dec3 < 255.
PSI	**Applicable only to Format 2.** Defines the Datanet as a passthrough with a profile DN7100/DN-C. As a passthrough, it must appear after the Local System and before all remote systems. In DSA addressing, it must also appear before any system referencing it through its PT parameter. ***For a passthrough, either PSI or ISL must be specified.*** *PSI=(CChexa2 [,CChexa2]):* Lists up to two PSI channels linking the Datanet to the Local System, each must be configured in the SRST, local or remote, of the targeted DPS 7000.
ISL alias **ISL1**	**Applicable only to Formats 1, 2 and 3**. Represents the first or single attachment of the system over an ISL cable. ***Mandatory for a neighbors. For a passthrough, either PSI or ISL must be specified.***
ISL2	**Applicable only to Formats 1, 2 and 3.** Represents the second attachment of the system over an ISL cable. ***ISL2 can only be specified if ISL1 is specified.***
ISL3	**Applicable only to Formats 1, 2 and 3.** Represents the third attachment of the system over an ISL cable. ***ISL3 can only be specified if ISL2 is specified.***

ISL4

Applicable only to Formats 1, 2 and 3.

Represents the fourth attachment of the system over an ISL cable.

ISL4 can only be specified if ISL3 is specified.

ISL parameters are used to declare:

– the local ISL communications paths used by the local TNS, when associated with the Local System (TYPE=LOCAL or PF=LSYS)

– the system as a passthrough, when TYPE is omitted, and PF=CNP7 or PF=DN7100

– the system as a neighbor, when TYPE is omitted and any other profile is specified.

A passthrough must be declared:

– *after the Local System and before all remote systems*

– *in DSA addressing, before any other system referencing it through its PT parameter.*

A neighbor must be declared before all remote systems.

ISL arguments:

The arguments of the ISL parameter are in the following order:

ISL address: **Applicable only to and mandatory for Formats 1, 2 and 3.** Must always be the first parameter. Represents the individual address of the transceiver on the ISL cable, of the format: *[{L|G}]*

DPS 7000: [08-00-38-] dh-hh-hh,

DPS 7000/XTA: hh-hh-hh-hh-hh-hh, no value reserved for BULL S.A. where:

– *L*: (local) or *G*: (global, ***default***) is network-wide and unique

– *08-00-38-* is optional and specific to the manufacturer

– 1 <= *d* <= 7, and *h* is hexadecimal.

ISL controller: **Applicable only to and mandatory for Format 1.** Identifies the local adaptor and must be registered in the SRST of the targeted system (local or remote), of the format: *EAhexa2*.

ISL cable-name: **Applicable only to Formats 1, 2 and 3.** Identifies the cable attachment corresponding to the current ISL parameter, of the format: *name8*.

ISL defaults: For the Local System, ***default*** for ISL1 (ISL) is CBL_MAIN.

For passthroughs and neighbors, the ***default*** for ISL1 is the corresponding value for ISL1 of the Local System.

The ***default*** for ISL2, ISL3 and ISL4 is the value for ISL1, ISL2 and ISL3, respectively.

PT alias **PASSTHRU** alias **PASS_THROUGH**

Applicable only to DSA Formats: *PT=(name4 [,name4 [...]])* where the functions of the passthrough(s) listed depends on the Format.

When applied to DSA Format 1:
Specifies all the passthroughs supporting network security control by which the Local System is *secured*. All such passthroughs must be declared by SYSTEM directives. No other passthroughs may be specified.

When applied to other DSA syntax formats:
Specifies all the passthroughs (up to 8 over a PSI channel and up to 8 over ISL) through which the current system can be accessed by the Local System. This list excludes passthroughs serving as front-ends for secondary networks.

The SYSTEM directive containing the PT parameter must appear after all the systems it lists have been declared.

Each object can be:
- an OCS Front End
- a passthrough system through which the current system can be accessed by the local system (up to 8 over a PSI channel and up to 8 over ISL).

NCC alias **NCC1, NCC2, NCC3, NCC4**	An ordered list of up to 4 OCS Front Ends attached to the system. The first part of the parameter is the individual address of the OCS Front End on the LAN. The address contains 17 characters of the format: `xx-xx-xx-xx-xx-xx` where `xx` are hexadecimal. This address may be prefixed by `L:` or `G:` – `L:` means local (not unique in the network). – `G:`, the default, means global (unique in the network). The second part of the parameter identifies the local OCS Front End as declared in the SRST. The format is EAxx.

NETWORK SECURITY CONTROL SERVICE

The site represented by the local system can be declared as *secured*. In this case security checking must be applied to all incoming connections from any remote system and also to any connection between local applications.

For security checks on incoming connections:

- The Local System must be declared *secured* in the network configurations of each passthrough front-ending the Local System and dedicated to its security
- The security server must be known and accessible by all the passthroughs front-ending the local system and dedicated to its security.

For security checks on local connections:

- The Local System must be locally declared *secured* by specifying NETSEC=YES in the SECOPT directive of the GCOS 7 configuration
- All the local connections must be internally transformed by the local session server (VCAM) into pseudo-incoming connections (from the local system towards itself, via a pseudo-remote system with reserved name BCLS
- In the PASSTHROUGH parameter of the SYSTEM directive describing the Local System, the list of only passthroughs dedicated to the access security for the local system must be specified. ***No other passthrough may be specified here.*** *No remote system explicitly named BCLS may exist in the basic network configuration.*

At enabling time, consistency between security option chosen at GCOS 7 system configuration and NETGEN network configuration is checked as follows:

- if security is set in GCOS 7 but no BCLS is declared in the basic network configuration, severity 4 error NG086 will be flagged
- if security is not requested in GCOS 7, no security checks will be performed over the local connections. The pseudo-system BCLS is meaningless in this case.

MUTUAL CROSS-REFERENCING

A special situation may occur when two passthroughs reference each other through the PT parameter. Since the first passthrough cannot reference the second, the **overriding mechanism** is used to resolve this anomaly. See Section 7.

POOLSZ alias **POOLSIZE**

Applicable only to Formats 1 and 2 (local or passthroughs). Specifies the buffer pool size in units of 64K bytes, for use by the appropriate communications server, as follows:

– TNS for the local system attached over the ISL
– an FEPS occurrence for a passthrough linked over the PSI.

In other cases, it does not apply.

POOLSZ=([min,] max), where:

min is the lower bound:
1 <= *min* <= 256 for FEPS and TNS
1 <= *min* <= 128 for OCS

max is the upper bound:
2 <= *max* <= 256 for FEPS and TNS
2 <= *max* <= 128 for OCS

min <= *max*, if only one value is given: *min* = *max*

***Default**:* (1,4) for FEPS and TNS.
***Default**:* (15,32) for OCS.

TCP alias **TCPIP**

Applicable only to Format 1 (Local System). Specifies the transport used:

TCP=0: ***Default****,* only ISO/DSA transport

TCP=1: TCP/IP transport optionally with ISO/DSA transport.
Using only TCP/IP transport allows the Local System to be the only system declared in the basic network configuration. No remote system can be described in the directory.

ISL must be specified if TCP is specified.

LOAD

Applicable only to Formats 2 and 3 (passthrough and neighbor). Specifies where the system is to be administered:

LOAD=0: ***Default for neighbors****,*

The system is not administered by the Local System but is instead remotely administered.

LOAD=1: ***Default for passthroughs****,*

The system is administered by the following local servers:
- FEPS and FECM if PSI defined (passthrough only)
- TNS and FECM if ISL defined (passthroughs and neighbors).

MADDR alias **MADDR1**
MADDR2, MADDR3, MADDR4

Lists up to 2 Multicast ADDResses.

MADDR=({L|G}:xx-xx-xx-xx-xx-xx,char19), where
- *L:* means local and *G:* means global
- *xx* are hexadecimal digits
- char19 represents {L|G}:xx-xx-xx-xx-xx-xx.

DPS 7000**:** ***Default****:* G:08-00-38-10-00-00, G:08-00-38-20-00-00, where

- G: the address is network-wide and unique
- 08-00-38 is reserved for BULL S.A. (DPS 7000 system only)

RADDR alias **RADDR1**
RADDR2, RADDR3, RADDR4

Lists up to 12 pair-sets of remote individual addresses. A pair-set represents a range of addresses, the first being the lower address, the second the upper. The addresses are separated by a /.

RADDR=(xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx), where

- *xx* are hexadecimal digits
- *d* takes even values: 0, 2, 4, 6, 8, A, C or E.

***Default**:* 00-00-00-00-00-00/7E-FF-FF-FF-FF-FF, where

- *08-00-38* is reserved for BULL S.A. (DPS 7000 system only)

OBJLIST alias **OBJECT_LIST Applicable to all Formats**.

OBJLIST=1: Lists for each SYSTEM directive, the set of expanded low-level directives immediately after this SYSTEM directive itself. Each entry expanded is flagged ->. This is useful when the overriding is used. See Section 7.

OBJLIST=0: ***Default***

Note:
Activating or deactivating the object list is effective from the current SYS directive to the end of the source or until another SYS directive defines a different value for OBJLIST.

OBJPFX alias **OBJECT_PREFIX** **Applicable to Syntax Format 1**.

OBJPFX=1 ***default****:* The names of the objects generated by a SYSTEM directive include a prefix specifying the type and possibly the generation rank of this object when several objects of the same type are attached to the same system (for more details, see Appendix B).

OBJPFX=0: When possible, that is, each time an object generated by the SYSTEM directive is unique for a given system, the name of the object is short (not prefixed). For more details, see Appendix B.

For example,

```
SYS FCOC PF=LSYS NCC=(10-00-11,EA81)
```

with OBJPFX = 1 (by default)

generates:

```
-->SVR NAME=OCS_EA81 TYPE=OCS
```

But,

```
SYS FCOC PF=LSYS NCC=(10-00-11,EA81) OBJPFX=0
```

with OBJPFX = 0 (specified explicitly)

generates:

```
-->SVR NAME=EA81 TYPE=OCS
```

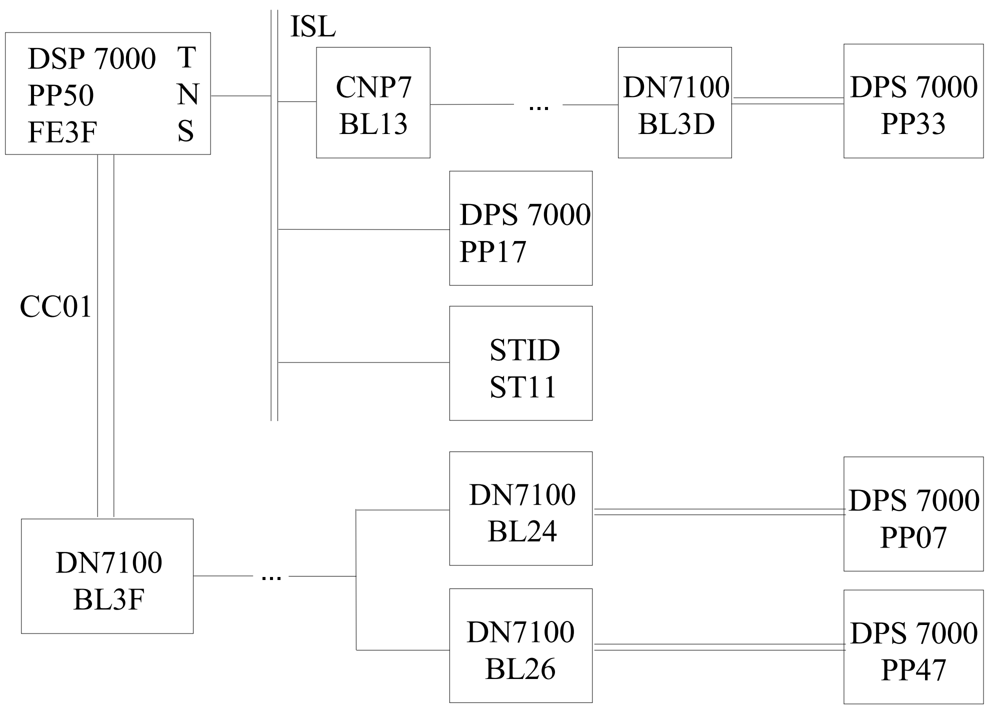
3.10 Network Configuration Description Examples

3.10.1 Example 1

I- NETWORK SCHEMATIC DESCRIPTION

The example shows:

- the local system PP50
- its neighbors
  - PP17 : direct DPS 7000-to-DPS 7000 link on ISL
  - ST11 : neighbor (DPX/2) on ISL
  - BL3F : passthrough (Datanet) on PSI
  - BL13 : passthrough (CNP 7) on ISL
- the remote systems of the primary network through either one of the passthrough.
- all the remote systems are configured in the basic network configuration. The directory is not used and will remain empty after such generation.
- the local system is seen as secured by the passthrough BL3F.



II- OSI NETWORK CONFIGURATION DESCRIPTION

Allows DSA applications to dialog with other DSA or ISO applications.

```
COMM '+--------------------------------------------+';
COMM '|      L O C A L  S Y S T E M  :  D P S 7    |';
COMM '+--------------------------------------------+';
SYS  PP50 TYPE=LOC SCID=60:57 ISL=(50-3C-39,EA11)
     PID=ISO  TSEL=50503530 NSAP=39250F080038503C39;

COMM '+--------------------------------------------+';
COMM '|                PASSTHROUGHS                |';
COMM '+--------------------------------------------+';
SYS  BL3F   PF=DN7100 SCID=60:63 PSI=CC01
            PID=ISO   NSAP=39250F060063;
SYS  BL13   PF='CNP7/CNS7/A0' PID=ISO ISL=(50-3C-13)
            PID=ISO   NSAP=39250F080038503C13 SCID=60:19;

COMM '+--------------------------------------------+';
COMM '|                 NEIGHBORS                  |';
COMM '+--------------------------------------------+';
DEF  PTPT17 TYPE=SET WATCH=2000 WINDOW=5;
SYS  PP17   PF=DPS7  TYPE=NB ISL=(50-3C-17)
            PID=ISO  NSAP=39250F080038503C13;
SYS  ST11   PF=STID  TYPE=NB ISL=(50-3C-11)
            PID=ISO  NSAP=39250F080038503C11;

COMM '+--------------------------------------------+';
COMM '|             REMOTE SYSTEMS                 |';
COMM '+--------------------------------------------+';
SYS  BL3D PF= DN7100            SCID=60:61 PT=BL13;
SYS  BL24 PF='DN7100/DN-C/V4.1' SCID=60:36 PT=BL3F;
SYS  BL26 PF='DN7100/DN-C/V4.1' SCID=60:38 PT=BL3F;
SYS  BP33 PF='DPS7/GCOS-7/V10'  PID=ISO NSAP=39250F060051;
SYS  PP07 PF='DPS7/GCOS-7/V10'  PID=ISO NSAP=39250F060007;
SYS  PP47 PF= DPS7              PID=ISO NSAP=39250F060071;

ENET;
```

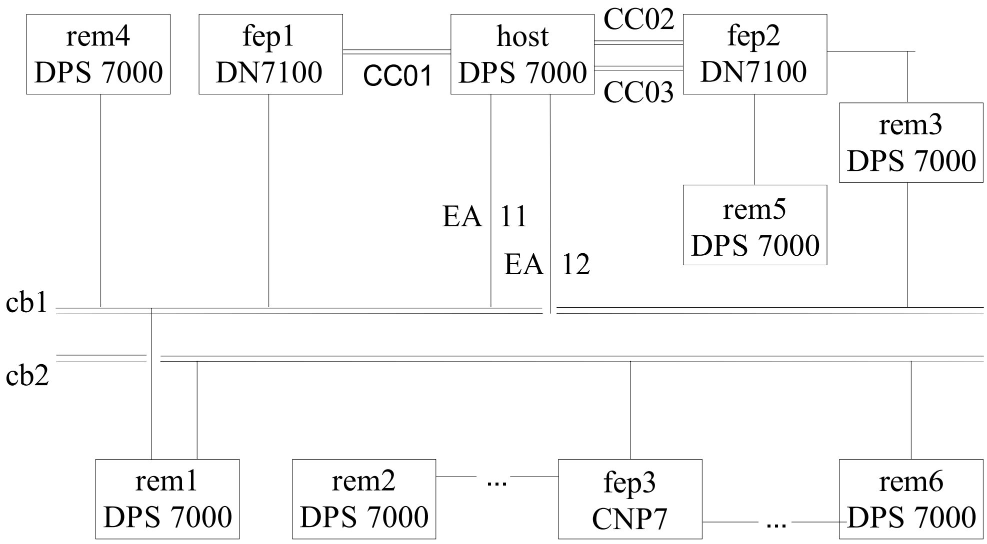
III- DSA NETWORK CONFIGURATION DESCRIPTION

Allows DSA applications to dialog with other DSA applications.

```
COMM '+----------------------------------------------+';
COMM '|     L O C A L  S Y S T E M  :  D P S 7     |';
COMM '+----------------------------------------------+';
SYS  PP50 TYPE=LOC SCID=60:57 ISL=(50-3C-39,EA11) PT=BL3F;

COMM '+----------------------------------------------+';
COMM '|                 NEIGHBOURS                   |';
COMM '+----------------------------------------------+';
SYS  BL13 PF='CNP7/CNS7/A0' SCID=60:19 ISL=(50-3C-13);
DEF  PTPT17 TYPE=SET        WATCH=2000 WINDOW=5;
SYS  PP17   PF=DPS7         SCID=60:23 ISL=(50-3C-17);
SYS  BL3F   PF=DN7100       SCID=60:63 PSI=CC01;

COMM '+----------------------------------------------+';
COMM '|             REMOTE SYSTEMS                   |';
COMM '+----------------------------------------------+';
SYS  BL3D PF= DN7100            SCID=60:61 PT=BL13;
SYS  BL24 PF='DN7100/DN-C/V4.1' SCID=60:36 PT=BL3F;
SYS  BL26 PF='DN7100/DN-C/V4.1' SCID=60:38 PT=BL3F;
SYS  BP33 PF='DPS7/GCOS-7/V10'   SCID=60:51 PT=BL13;
SYS  PP07 PF='DPS7/GCOS-7/V10'   SCID=60:07 PT=BL3F;
SYS  PP47 PF= DPS7              SCID=70:71 PT=BL3F;

ENET;
```

3.10.2 Example 2

I- NETWORK SCHEMATIC DESCRIPTION



II- OSI NETWORK CONFIGURATION DESCRIPTION

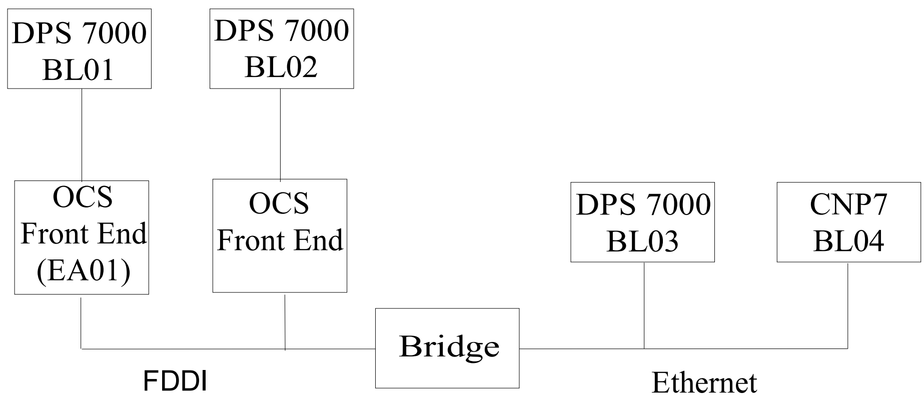
```
COMM '+--------------------------------------------+';
COMM '|         B A S I C   N E T W O R K          |';
COMM '+--------------------------------------------+';

SYS host, TYPE=LOC, PID=ISO, SCID=45:123, TSEL=484F5354,
          NSAP=(39250F080038123456, 39250F080038234567),
          ISL =(12-34-56,EA11,cb1), ISL2=(23-45-67,EA12,cb2);

SYS fep1, TYPE=PT, PF=DN7100, SCID=46:122,
          ISL =(34-56-78,cb1) PSI=CC01;
DEF       TYPE=SET, WATCH=222;
SYS fep2, TYPE=PT, PF=DN7100, SCID=47:121,
          PSI =(CC02,CC03), POOLSZ=(2,5);
DEF       TYPE=SET, SILENT=111;
SYS fep3, TYPE=PT, PF=CNP7,
          ISL1=(56-78-9A,cb2), SCID=48:120;

SYS rem1, TYPE=NB, PID=ISO, PF=DPS7,
          NSAP=(39250F080038456789, 39250F080038789ABC)
          ISL1=(45-67-89,cb1), ISL2=(78-9A-BC,cb2);
DEF       TYPE=RESET, CLASS=TS;
SYS rem3, TYPE=NB, PID=ISO, PF=DPS7,
          NSAP=39250F080038325476,
          ISL =(32-54-76,cb1);
SYS rem4, TYPE=NB, PID=ISO, PF=DPS7,
          NSAP=39250F08003879ABCD,
          ISL1=(79-AB-CD,cb1);
SYS rem6, TYPE=NB, PID = ISO, PF = DPS7,
          NSAP=39250F0800387ABCDE,
          ISL =(7A-BC-DE,cb2);

COMM '+--------------------------------------------+';
COMM '|             D I R E C T O R Y              |';
COMM '+--------------------------------------------+';

DIR;
SDOM      NSAP_PREFIX=39250F0      NB=(fep2, rem3);
SDOM      NSAP_PREFIX=39250F080038 NB=(fep1:1:4);
SDOM      NSAP_PREFIX=39250F   NB=(fep3:2, rem6:3:5, rem1:0);

SYS rem2, TYPE=RMT, PID=ISO, PF=DPS7,
          NSAP=39250F043119;
SYS rem5, TYPE=RMT, PID=ISO, PF=DPS7,
          NSAP=39250F040116;
EDIR;
```

III- DSA NETWORK CONFIGURATION DESCRIPTION

```
COMM '+-------------------------------------------+';
COMM '|        B A S I C   N E T W O R K          |';
COMM '+-------------------------------------------+';

SYS host, TYPE=LOC, SCID=45:123,
          ISL =(12-34-56,EA11,cb1),
          ISL2=(23-45-67,EA12,cb2);

SYS fep1, PF  =DN7100, SCID=46:122,
          ISL =(34-56-78,cb1)
          PSI =CC01;
DEF       TYPE=SET, WATCH=222;
SYS fep2, PF  =DN7100, SCID=47:121,
          PSI =(CC02,CC03), POOLSZ=(2,5);
DEF       TYPE=SET, SILENT=111;
SYS fep3, PF  =CNP7, SCID=48:120,
          ISL1=(56-78-9A,cb2);

SYS rem1, PF  =DPS7, SCID=44:120,
          ISL1=(45-67-89,cb1),
          ISL2=(78-9A-BC,cb2);
DEF       TYPE=RESET, CLASS=TS;
SYS rem3, PF  =DPS7, SCID=42:118,
          ISL =(32-54-76,cb1),
          PT  =fep2;
SYS rem4, PF  =DPS7, SCID=41:117,
          ISL1=(79-AB-CD,cb1)
          PT  =fep1;
SYS rem6, PF  =DPS7, SCID=39:115,
          ISL =(7A-BC-DE,cb2)
          PT  =fep3;

COMM '+-------------------------------------------+';
COMM '|            D I R E C T O R Y              |';
COMM '+-------------------------------------------+';

DIR host;

SYS rem2, PF=DPS7, SCID=43:119, PT=fep3;
SYS rem5, PF=DPS7, SCID=40:116, PT=(fep2);
EDIR;
```

3.10.3 Example 3 (FCP7 Support)

I- NETWORK SCHEMATIC DESCRIPTION



II DSA NETWORK CONFIGURATION DESCRIPTION (NETGEN Description)

Allows DSA applications to dialog with other DSA applications.

```
COMM '+-----------------------------------+';
COMM '!          LOCAL SYSTEM : BL01        !';
COMM '+-----------------------------------+';
SYS NAME=BL01 TYPE=LOCAL SCID=60:01 NCC1=(08-00-38-10-00-01,EA01);

COMM '+-----------------------------------+';
COMM '!           NEIGHBOR SYSTEMS          !';
COMM '+-----------------------------------+';
SYS NAME=BL02 TYPE=REMOTE PF=DPS7 SCID=60:02 PT=(EA01);
SYS NAME=BL03 TYPE=REMOTE PF=DPS7 SCID=60:03 PT=(EA01);
SYS NAME=BL04 TYPE=REMOTE PF='CNP7/CNS/A1' SCID=60:04 PT=(EA01);
```

NOTE:

In addition to the NETGEN declarations, each remote system must be declared in the OCS Front End OSI/DIWS configuration subfile using the NRSC and NRTS directives. See Section 8.

BL03 administers the CNP7/BL04. BL01 cannot administer the CNP7/BL04.

4. Network Administration

4.1 Scope of Network Administration

Generating the network administration enables the following actions :

- Flag local system unsolicited events such as Open Session's and NAD errors.
- Store these events in local logfiles.
- Send them to remote Administrative Correspondents throughout the network.
- Store unsolicited events produced by remote systems in local logfiles.
- Report events to local operators.
- Execute on the local system commands issued by remote Administrative Correspondents throughout the network.
- Send by local system commands to be executed by remote Administrative Correspondents.
- Emit History Reports on connections periodically.

4.1.1 Network Administration Objects

The following DSAC objects are used to administer the network:

Administrative Function (AF)

An AF is a subsystem that manages administrative information exchanged with both local and remote Administrative Correspondents. The three AFs defined for the DPS 7000 are:

- the **local NAD** (LNAD) which executes administrative commands,
- the **local MUX** (LNOI) which sends commands to remote administrative correspondents and reports events to operators,
- the **local logfile** handler (DSALOG).

Administrative Correspondent (AC)

An AC is always the local representation of a remote (or sometimes a local) Administrative Function: so defining the AC itself as being respectively either remote or local.

An AC belongs always to an Administrative Group having the same type as the AC and through which it is functionally associated to one of the three local AFs.

The following ACs apply for the DPS 7000:

- the local AC LDSALOG representing the local LNAD AF which is in functional relation with the local DSALOG AF,
- the ACs which are in functional relation with the LNAD AF and which can be of the following types :
  - NAD : AC representing remote NADs,
  - AUT : AC representing remote AUPI applications,
  - NOI : AC representing remote NOIs,
  - NCC : AC representing remote NCCs,
  - ASF : AC representing remote DSALOGs.
- the ACs which are in functional relation with the LNOI AF and which can be of the following types:
  - NAD : representing NADs located on remote heterogeneous systems,
  - MUX : representing MUX function located on remote homogeneous systems.

Administrative Group (AG)

An AG is a set of Administrative Correspondents. The following AGs may be specified:

- **RLNAD** composed of dynamically created remote ACs associated to the LNAD administrative function.They represent remote or local AUTs or remote NOIs or NCCs dialoging with the local NAD. The following filters can be attached to the AG:
  - input filters for selecting incoming commands,
  - output filters for selecting UMs to be sent.
- **RDSALOG** composed of dynamically created remote ACs associated to the DSALOG administrative function.They represent remote NADs which log messages into the log files of the local system. No filter can be attached to this AG.
- **RLNOI** composed of statically configured or -dynamically created remote ACs associated to the LNOI administrative function. They represent remote heterogeneous systems receiving commands or messages in AEP format from the local system.
  Input filters can be attached to this AG for the selection of incoming commands.
  Output filters can be attached to this AG for the selection of UMs to be sent.
- **RMUX** composed of statically configured or dynamically created remote ACs associated to the LNOI administrative function. They represent remote homogeneous systems exchanging commands or messages in GCOS format with the local system.

  No filter can be attached to this AG.
- **LNOI** composed of dynamic OPs associated with the LNOI administrative function. They represent the image of remote or programmed operators located in the local system.
- **other named AGs** corresponding to statically configured ACs which can be of the following types:
  - **NCC** or **NOI**:
    They represent emission of commands from remote NCCs (DPS6) or NOIs towards the local NAD and backwards sending of responses and messages to them from the local NAD. Input or output filters can be attached to this types of AG.
  - **ASF**:
    They represent logging of commands or messages from the local system into log files located on remote systems. Output filters can be attached to this type of AG.

– **AUT**:
They represent emission of commands from local or remote administrative utilities towards the local NAD and backwards receiving responses and messages from the local NAD. The following filters can be attached to the AG:
input filters for selecting incoming commands,
output filters for selecting UMs to be sent.

Administrative Log (LG)

An LG is a set of files managed by the DSALOG function. A maximum of two set can be defined. Output filters can be attached to each set of files.

Administrative Filter (FL)

An FL defines a set of criteria to selectively regulate the flow of administrative traffic applied to an AC, an AG or a LG.

Statistics Block (SB)

By declaring a Statistics Block object and specifying its parameters, one selects:

- a set of connection objects whose related History Report (HR) messages will be emitted periodically
- and the emission rate of these HR messages.

A connection object is said monitored when the periodic emission of its HR message is controlled by means of a Statistics Block object.

An SB controls the emission rate of history reports on the following connection object classes:

- channel connection (CC),
- ISO session connection (IS),
- logical connection (LC),
- physical connection (PC),
- DSA session connection (SS),
- transport connection (TC).

More precisely an SB defines a set of connections, belonging to the same class, whose related history reports are emitted periodically for the duration of the connection.

Monitoring connection objects using SBs, require the following sequential steps:

- **Declaring the SB Object:**
  The network administrator has to know what connection objects require monitoring by the SBs so that the relevant set of SB directives can be inserted in the basic network configuration description.
- **Generating the SB Object**
  SBs are generated by the NETGEN utility from the SB directives either explicitly declared or by default. SBs are available to monitor their connection objects as soon as the basic network configuration is enabled.
- **Handling SB Monitoring**
  To start an SB monitoring session, enter the STSVR operator command which activate the RAEH server.

When this is done, only the connection objects related to SBs in the ENABLE state will be actually monitored and HR messages will be emitted periodically for them.

The network administrator can display and modify some SB attributes using operator commands, irrespective of whether the RAEH server is started. The SB state can be modified to start or stop the emission of HR messages. The emission rate of HR messages can also be modified.

SBs generated by default are always initiated in the LOCK state so that connection objects related to them cannot be start to be monitored without operator intervention to set them in the ENABLE state.

When the RAEH server is terminated through the TTSVR operator command, no SBs remain active to monitor connection objects.

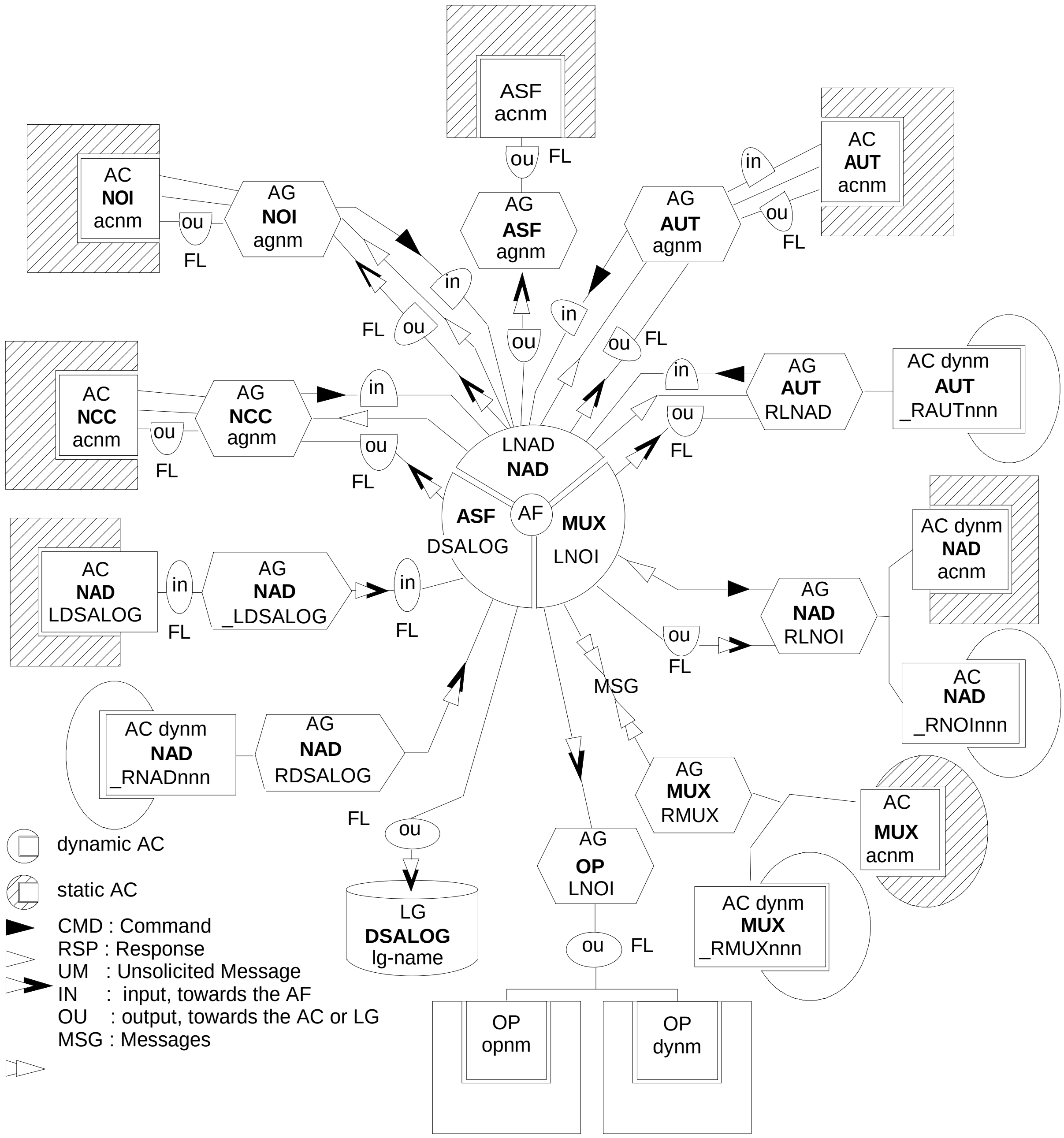


Figure 4-1. Schema of DSAC Objects

4.1.2 Network Administrative Default Configuration

The default network administrative configuration enables using standard administrative functions without having to configure a user-defined administration for a given network.

This default configuration is a set of network administration directives with values typical of most DPS 7000 networks. The user can redefine any of these directives with values more suited to the needs of a specific network.

These default functions are as follows:

- Unsolicited events sent by remote systems are stored in the local SYS.DSALOGi log files.
- Commands sent by administrative correspondents located on any system in the network, including the local system, are executed.
- Commands from heterogeneous systems (AEP format) are concurrently executed and their responses are sent.
- Commands are sent towards heterogeneous systems (AEP format) for which their responses are awaited.
- Messages are sent towards homogeneous systems (GCOS format).

The following default limits apply:

- The general maximum enqueuing size is 64.
- A timeout is provided to free idle sessions between dynamic ACs and AF after 30 minutes.
- Up to 32 ACs of an AG may concurrently establish a session with a local AF, see Section 10 for detailed limits.

4.2 Network Administration Directives

General Syntax Rules

Within the source corresponding to the basic network configuration the network administration directives may be entered in any order between the directive describing the last remote system and the ENET directive which must be the last.

The sequence in which NETGEN directives are declared does not affect the access time for any object but it does affect the order in which the objects are listed.

The DNET command lists network objects:

- either in the same order as they were declared if these objects concern TNS
- or in the reverse order for directory objects, systems and their session controls.

The following schema shows the sequence for declaring a typical basic network configuration description. See Section 3 for the description and syntax of SYSTEM directives.

```
[ NETWORK ]

  local system

  passthrough
   .
   .
   .
  neighbor
   .
   .
   .
  remote system
   .
   .
   .
[ network administration directive ]
   .
   .
   .
[ ENDNETWORK ]
```

NOTES:

1. For all directives, parameters can be specified in any order. All arguments are introduced by their corresponding keyword. NAME is the only exception being the first positional and can therefore be omitted.
2. In any directive when the argument is boolean (0 or 1), the keyword on its own is equivalent to specifying 1 as its argument, for example:
   BACKUP is equivalent to BACKUP=1
   NBACKUP is equivalent to BACKUP=0.

List of Network Administration Directives

In this Section the NETGEN directives are described in the alphabetical order of their abbreviations. The list below puts the same directives in alphabetical order of their expanded acronyms, giving also the corresponding abbreviation, configuration description destination (B for Basic) and paragraph number.

Abbr	Expanded	Dest	Par.	Abbr	Expanded	Dest	Par.
AC	ADMCOR	B	4.3	AG	ADMGROUP	B	4.5
FL	ADMFILTER	B	4.6	LG	ADMLOG	B	4.7
AF	ADMFUNC	B	4.4	SB	STATISTICS	B	4.8

4.3 AC (ADMCOR) Directive

Purpose

This directive applies only to the basic network configuration description.

It statically configures some Administrative Correspondents which are local representations of some remote AFs located on remote systems or of local or remote administrative utilities. *The administrative correspondents which can be only dynamically created do not apply to this directive.* Administrative correspondents are always grouped within AGs of same type.

Administrative filters may be associated to each AC. See FL directive. The default filters for an AC are those attached (explicitly or by default) to the AG to which it belongs. Likewise when a parameter can be specified in either the AC or the AG directive, the default value for this parameter in the AC directive is that attached either explicitly declared or configured by default, to the AG to which it belongs.

Syntax Format 1

```
{ AC     }
{        }          NAME = name8
{ ADMCOR }

                {         [         { ASF } ] }
                { name-8 [,TYPE =  { AUT } ] }
 { AG       }   {         [         { NCC } ] }
,{          } = {         [         { NOI } ] }
 { ADMGROUP }   {                             }
                { RLNOI  [,TYPE =    NAD   ] }
                {                             }
                { RMUX   [,TYPE =    MUX   ] }

 { SC   }
,{      } = name4
 { SESS }

 { MBX     }   { name8  }  [  { EXT    }          ]
,{         } = {        }  [, {        } = name4 ]
 { MAILBOX }   { $name7 }  [  { MBXEXT }          ]

[  { BACK   }   { 0 } ]
[, {        } = {   } ]
[  { BACKUP }   { 1 } ]
```

```
[  { ST    }   { ENBL } ]
[, {       } = {      } ]
[  { STATE }   { LOCK } ]

[  { CODE      }   { ASCII  } ]
[, {           } = {        } ]
[  { CHAR_CODE }   { EBCDIC } ]

[, MAXEV = dec3 ]

[, MAXCMD = dec3 ] ;
```

Syntax Format 2

```
{ AC     }
{        }       NAME = LDSALOG
{ ADMCOR }

[  { ST    }   { ENBL } ]
[, {       } = {      } ] ;
[  { STATE }   { LOCK } ]
```

Parameter Description

NAME Identifies the Administrative Correspondent:

NAME=LDSALOG: applicable to Format 2, represents the local LNAD AF. **When not specified, is automatically generated by the system.**

NAME=name8: applicable to Format 1; Name cannot start with an underscore as such naming is reserved for object dynamically generated by the system. No default.

AG alias **ADMGROUP** Identifies the Administrative Group to which the AC belongs:

AG=name8: **Not applicable to Format 2 AC LDSALOG**
ACs associated with AG RDSALOG and AG RLNAD cannot be explicitly declared since both are created dynamically. No default.

AG=RLNOI, AG=RMUX: ACs normally associated with AG RLNOI and AG RMUX need not be declared since these ACs are dynamically created and named by the NAD upon each connection request.

The naming convention used by the NAD for the dynamically created ACs is as follows:
- _RNAD*nnn* for AG RDSALOG
- _RAUT*nnn* for AG RLNAD
- _RNOI*nnn* for AG RLNOI
- _RMUX*nnn* for AG RMUX

where *nnn* is 3 decimal digits.

SC alias **SESS**

Identifies the Session Control of System in which the AC *resides*:

- either the local system defined in the LSC directive
- or remote system defined in a RSC directive.

SC=name4: **Not applicable to Format 2 AC LDSALOG**
No default.

MBX alias **MAILBOX**

Not applicable to Format 2 AC LDSALOG.
MBX=name8: Identifies the mailbox associated with the AC.
Optionally starts with $. No default.

EXT alias **MBXEXT**

Not applicable to Format 2 AC LDSALOG.
EXT=name4: Identifies the mailbox extension.
Default is 4 spaces.

BACK alias **BACKUP**

Not applicable:
- **to Format 2 AC LDSALOG**
- **and if the AC belongs to AG RLNOI or AG RMUX.**

Meaningful only if the AC is declared with INITIATOR=AF.

BACK=0: ***Default***
The AC is primary. If one or more backup ACs are declared for the same AG then only one primary AC can be declared.

BACK=1: The AC is backup. Such ACs should be declared after the primary AC. The order in which these ACs are declared will determine the order in which the administrative server will dynamically scan them.

ST alias **STATE**

Defines the initial state of the AC.

ST=ENBL: ***Default***
The AC is operational at the start of the communications session.

ST=LOCK: The AC waits to be set operational by the operator.

MAXEV

Not applicable to Format 2 AC LDSALOG.
Specifies the maximum number of concurrent log events waiting to be sent to the AC:

- when specified, overrides for the current AC, the MAXEV value of the owner AG
- when omitted the default is the MAXEV value of the owner AG.

1 <= *maxev* <= 511.

MAXCMD

Not applicable to Format 2 AC LDSALOG.

Specifies the maximum number of:
- commands waiting to be sent
- commands awaiting responses
- concurrent input commands
- responses waiting to be sent.

1 <= *maxcmd* <= 255.

For AG RMUX, MAXCMD is the maximum number of concurrent messages waiting to be sent in the case of homogeneous systems:
- when specified, overrides for the current AC, the MAXCMD value of the owner AG
- when omitted, the default is the MAXCMD value of the owner AG.

CODE alias **CHAR_CODE**

Not applicable to Format 2 AC LDSALOG. Defines the character set code used by the AC.
CODE=ASCII: ***Default***
CODE=EBCDIC

TYPE

Specifies the type of the static AC:

TYPE=ASF: ***Default if AG is neither RLNOI nor RMUX***, Remote ASF logging:
- messages from the local system are journalized in logfiles located on the remote system.
- Output filters can be attached to this AC for selecting messages to be logged.

TYPE=AUT: Represents a remote AUT sending commands and UMs to the local LNAD AF:
- An AC which is not declared and which dialogs with the local AF LNAD, is considered type AUT belonging to AG RLNAD
- Statically declared ACs of type AUT cannot be attached to AG RLNAD but instead to a user-defined AG of type AUT
- Input filters can be attached to this AC for selecting incoming commands
- Output filters can be attached to this AC for selecting the UMs to be sent.

TYPE=MUX: ***Default if AG is RMUX***, Represents an NOI on a remote homogeneous system receiving commands or messages in GCOS format from the local system:
- Such an AC can only be grouped under AG RMUX
- No filters are allowed for these ACs.

TYPE=NAD: ***Default if AG is RLNOI***, Represents a NAD on a remote heterogeneous system receiving commands or messages in AEP format from the local system:
- Such an AC can only be grouped under AG RLNOI
- No filters are allowed for these ACs.

TYPE=NCC: Represents a remote NCC on a DPS 6 or an NOI sending commands to, and receiving in return responses and messages from the local AF LNAD:
- Output filters can be attached to this AC for selecting the UMs to be sent.

TYPE=NOI: Represents a remote NOI sending commands to, and receiving in return responses and messages from the local AF LNAD:
- Output filters can be attached to this AC for selecting the UMs to be sent.

4.4 AF (ADMFUNC) Directive

Purpose

This directive applies only to the basic network configuration description. The AF directive defines an administrative function and its use by ACs located in either the same local system or remote systems.

Syntax Format 1

```
{ AF      }
{         }       NAME = LNAD ;
{ ADMFUNC }
```

Syntax Format 2

```
{ AF      }                             [ { ST    }   { ENBL } ]
{         }       NAME = DSALOG         [,{       } = {      } ] ;
{ ADMFUNC }                             [ { STATE }   { LOCK } ]
```

Syntax Format 3

```
{ AF      }
{         }       NAME = LNOI ;
{ ADMFUNC }
```

Parameter	Description
NAME	Identifies the Administrative Function. *NAME=LNAD:* Applicable to Format 1 Executes commands in AEP format received from ACs of types: NAD, AUT, NOI or NCC. **When not specified, is automatically generated.** *NAME=DSALOG:* Applicable to Format 2 Accepts unsolicited messages (events) from local and remote systems, and journalizes them in the local DSALOG file being an AC ASF. **When not specified, is automatically generated.** *NAME=LNOI:* Applicable to Format 3 Exchanges commands and responses in AEP format with AC NADs located on remote heterogeneous systems *and* messages in GCOS format with AC MUXs located on remote homogeneous systems. **When not specified, is automatically generated.**
ST alias **STATE**	Defines the initial state of the AF: *ST=ENBL: **Default for all AFs*** The AF is operational at the start of the communications session. *ST=LOCK:* The AF awaits operator intervention to function.

4.5 AG (ADMGROUP) Directive

Purpose

This directive applies only to the basic network configuration description. The AG directive groups together a set of ACs having the same type as the AG and also having common operational attributes with regard to an AF. An AC can belong to only one AG. ACs on different systems can be grouped under a single AG.

Default AFs are defined for some types of AG. They are effective only if not overridden by any other filter explicitly attached to the AG or to one of the ACs belonging to this AG. Likewise, for parameters common to the AG and AC directives, the value specified (explicitly or by default) in the AG applies to all its ACs if not overridden by a value explicitly specified in the individual ACs themselves.

Syntax Format 1

```
{ AG       }                               [          { NCC } ]
{          }         NAME = name8          [, TYPE = { ASF } ]
{ ADMGROUP }                               [          { NOI } ]
                                           [          { AUT } ]
{ AF       }   { LNAD   }
{          } = {        }
{ ADMFUNC  }   { DSALOG }

[   { ST    }   { ENBL } ]
[,  {       } = {      } ]
[   { STATE }   { LOCK } ]

[ { INIT      }   { AC | ADMCOR  } ]
[,{           } = {              } ]
[ { INITIATOR }   { AF | ADMFUNC } ]

[ { MAXSC    }   { 32   } ]
[,{          } = {      } ]
[ { MAXSESS  }   { dec3 } ]

[   { N        }   { 0    } ]
[,  { RETRY    } = {      } ]
[   { MAXRETRY }   { dec5 } ]

[   { T1         }   { 10   } ]
[,  { FLOW       } = {      } ]
[   { RETRYDELAY }   { dec3 } ]
```

```
[   { T2        }   { 50   } ]
[, { SILENT    } = {      } ]
[   { IDLEDELAY }   { dec3 } ]

[             { 64   } ]
[, MAXCMD = {      } ]
[             { dec3 } ]

[            { 64   } ]
[, MAXEV = {      } ] ;
[            { dec3 } ]
```

Syntax Format 2

```
{ AG        }
{           }   NAME = RDSALOG
{ ADMGROUP }

[   { MAXSC   }   { 32   } ]
[, {         } = {      } ] ;
[   { MAXSESS }   { dec3 } ]
```

Syntax Format 3

```
{ AG        }
{           }   NAME = RLNAD
{ ADMGROUP }

[   { MAXSC   }   { 32   } ]
[, {         } = {      } ]
[   { MAXSESS }   { dec3 } ]

[   { IDLEDELAY }   { 50   } ]
[, { SILENT    } = {      } ]
[   { T2        }   { dec3 } ]

[             { 64   } ]
[, MAXCMD = {      } ]
[             { dec3 } ]

[            { 10   } ]
[, MAXEV = {      } ] ;
[            { dec3 } ]
```

Syntax Format 4

```
{ AG       }
{          }   NAME = RLNOI
{ ADMGROUP }

[  { MAXSC   }   { 32   } ]
[, {         } = {      } ]
[  { MAXSESS }   { dec3 } ]

[              { 4    } ]
[,RESPDELAY = {      } ]
[              { dec3 } ]

[  { T2        }   { 50   } ]
[, { SILENT    } = {      } ]
[  { IDLEDELAY }   { dec3 } ]

[            { 64   } ]
[, MAXCMD = {      } ]
[            { dec3 } ]

[           { 10   } ]
[, MAXEV = {      } ] ;
[           { dec3 } ]
```

Syntax Format 5

```
{ AG       }
{          }   NAME = RMUX
{ ADMGROUP }

[  { MAXSC   }   { 32   } ]
[, {         } = {      } ]
[  { MAXSESS }   { dec3 } ]

[  { T2        }   { 50   } ]
[, { SILENT    } = {      } ]
[  { IDLEDELAY }   { dec3 } ]

[            { 64   } ]
[, MAXCMD = {      } ] ;
[            { dec3 } ]
```

Parameter	Description
NAME	Identifies the Administrative Group. *NAME=name8:* Applicable to Format 1 – Underscore (_) as first character is reserved for AGs created by the system – Depending on their type, input or output filters may be attached to these AGs. Default filters are also supplied, see TYPE. *NAME=RDSALOG:* Applicable to Format 2 – Groups only remote dynamic AC NADs which send UMs for journalizing in the logfiles of the local system; no filters are allowed for this AG – **When not specified, is automatically generated with defaults.** *NAME=RLNAD:* Applicable to Format 3 – Groups only remote dynamic AC AUTs or AC NCCs sending commands and unsolicited messages to the local AF LNAD – Input filters can be attached to this AG for selecting incoming commands – When no input filter is specified, two default input filters are generated which exclude updating commands. See FL directive) – Output filters can be also attached to this AG for selecting unsolicited messages to be sent – When no output filter is specified, two default output filters are generated which inhibit the sending of all unsolicited messages, see FL directive – **When not specified, is automatically generated with defaults.**

NAME=RLNOI: Applicable to Format 4

- Groups static and dynamic AC NADs on remote heterogeneous systems which exchange commands and responses, or unsolicited messages in AEP format with the local system
- Input filters can be attached to this AG for selection of the incoming commands
- When no input filter is specified, a default input filter is generated which accepts all commands, see FL directive
- No output filters can be attached to this AG
- **When not specified, is automatically generated with defaults.**

NAME=RMUX: Applicable to Format 5

- Groups static and dynamic ACs being AF LNOIs on remote homogeneous systems which exchange in GCOS format commands, responses or unsolicited messages with the local system
- No filter allowed for this AG.
- **When not specified, is automatically generated with defaults.**

TYPE

Applicable to Format 1, for determining the type of associated ACs.

TYPE=NOI
TYPE=NCC: ***Default***, Groups remote static AC NCCs or AC NOIs which send commands to, and receive in return responses and unsolicited messages from the local LNAD AF:

- Input filters can be attached to these AGs for selecting incoming commands
- When no input filter is specified, a default input filter is generated which accepts all commands, see FL directive
- Output filters can be attached to these AGs for selection of the UMs to be sent
- When no output filter is specified, a default output filter is generated which allows sending all unsolicited messages whose importance level >= 4, see FL directive.

TYPE=ASF: Groups remote static AC ASFs which journalize commands and unsolicited messages from the local system in logfiles located on the remote systems:
- Output filters can be attached to these AGs for selecting the unsolicited messages to be sent
- When no output filter is specified; two default output filters are generated which allow sending all unsolicited messages whose importance level >= 12 and those pertaining to administrative domains 2 or 3. See FL directive.

TYPE=AUT: Groups remote static AC AUTs using AUPI, which send commands to, and receive in return responses and unsolicited messages from the local AF LNAD:
- Input filters can be attached to these AGs for selecting incoming commands
- When no input filter is specified, two default input filters are generated which exclude all updating command. See FL directive
- Output filters can be attached to these AGs for selecting the unsolicited messages to be sent
- When no output filter is specified, two default output filters are generated which inhibit sending unsolicited messages. See FL directive.

AF alias **ADMFUNC**

Identifies the Administrative Function associated with the AG. GCOS-7 has 3 AFs:

AF=LNAD:	AF NAD
AF=DSALOG:	AF ASF
AF=LNOI:	AF MUX (reserved internal value)

ST alias **STATE**

Defines the initial state of the AG:

ST=ENBL: ***Default***
The AG is operational at the start ofd the communications session.

ST=LOCK: **Applicable only to the Format 1 AGs**.
The AG awaits operator intervention to be functional.

INIT alias **INITIATOR**

Specifies the Initiator of connections:

INIT=AF: **Applicable only to the Format 1 AGs**.
AF is initiator. At least one AC belonging to the AG must be declared primary. Although several backup ACs may be declared, only one primary is allowed and must be the first declared. Where backup ACs are declared, the MAXSESS value is internally forced to 1. See AC directive.

INIT=AC: **Applicable to and is the *Default* for Formats 1, 2 and 3 AGs**.
ACs of this AG are initiator.

INIT=BOTH: **Applicable to and is the *Default* for Formats 4 and 5 AGs**.
Either AF or AC can be initiator.

MAXSC alias **MAXSESS**

Maximum number of sessions which can be established at one time with the ACs owned by this AG:

- For AG RDSALOG (Format 2) related sessions are towards the local AF DSALOG.
- For AG RLNAD (Format 3) related sessions are towards the local AF LNAD.
- For AG RLNOI (Format 4) related sessions are between the local AF LNOI and ACs on heterogeneous systems (mailbox $NAD).
- For AG RMUX (Format 5) related sessions are between the local AF LNOI and ACs on homogeneous systems (mailbox $MUX).

1 <= *maxsess* <= 420. ***Default***: 32.

When INITIATOR=AF and backup ACs are declared, MAXSESS is reduced to 1.

N or **RETRY** alias **MAXRETRY**

Applicable only if INITIATOR=AF.

Maximum number of retries to open a connection with one of the ACs in the AG.

0 <= *retry* <= 10000. ***Default:*** 0 for infinite count.

T1 or **FLOW** or **RETRYDELAY**

Applicable only if INITIATOR=AF.
Maximum delay in hundredths of an hour between possible successive retries for opening a connection with one of the ACs in the AG.

1 <= *flow* <= 100. ***Default:*** 10.

RESPDELAY

Applicable only to AG RLNOI (Format 4).
Maximum delay in hundredths of an hour between the emission of a command to a heterogeneous system and the reception of its response.

1 <= *respdelay* <= 255. ***Default:*** 4.

T2 or **SILENT** or **IDLEDELAY**

Not applicable to either AF=DSALOG or AF=LNAD specified with INIT=AF.
Maximum delay in hundredths of an hour for closing the connection because of no activity.
IDLEDELAY=0 means no time-out delay.

0 <= *idledelay* <= 255. ***Default:*** 50.

MAXCMD

Not applicable to AF=DSALOG.
Specifies the type of maximum traffic to be handled by *each AC of the AG*:

- For Format 1, 3 and 4 AGs: the maximum number of concurrent input commands or the maximum number of responses waiting to be sent.
- For AG RLNOI (Format 4): the maximum number of commands waiting to be sent to, or the maximum number of commands awaiting responses from heterogeneous systems.
- For AG RMUX (Format 5): the maximum number of concurrent messages waiting to be sent to homogeneous systems.

1 <= *maxcmd* <= 255. ***Default:*** 64.

MAXEV

Not applicable to AF=DSALOG and to AG RMUX (Format 5).
Specifies for each AC of the AG, the maximum number of concurrent log events waiting to be sent.

1 <= *maxev* <= 511.

Default: 10 if AF=RLNAD or AF=RLNOI, 64 otherwise.

4.6 FL (ADMFILTER) Directive

Purpose

This directive applies only to the basic network configuration description. It allows selection of the flow of administrative traffic between AFs and ACs. They may be attached either to ACs, AGs or the sets of logfiles (LOG).

When not explicitly declared, default filters are automatically generated for the set of logfiles and some types of AGs.

Syntax

```
{ FL        }                                    { { IN   | INPUT }  }
{           }        NAME = name8     ,TYPE = {                    }
{ ADMFILTER }                                    { { OUT | OUTPUT } }

 { { AC | ADMCOR   }          }
,{ { AG | ADMGROUP } = name8 }
 { { LG | ADMLOG   }          }

[            { INCL | INCLUSIVE } ]
[, LOGIC = { EXCL | EXCLUSIVE } ]
[            { MAND | MANDATORY } ]

[   { ST    }   { ENBL } ]
[, {        } = {      } ]
[   { STATE }   { LOCK } ]

[   { DOM    }          [   { 4    } ]   ]
[, {         } = ( dec1 [, {       } ] ) ]
[   { DOMAIN }          [   { dec1 } ]   ]

[   { PWR   }          [   { 3    } ]   ]
[, {        } = ( dec1 [, {       } ] ) ]
[   { POWER }          [   { dec1 } ]   ]

[                      [   { 255  } ]   ]
[, CLASS = ( dec3 [, {       } ] ) ]
[                      [   { dec3 } ]   ]

[   { CODE   }          [   { 255  } ]   ]
[, {         } = ( dec3 [, {       } ] ) ]
[   { OPCODE }          [   { dec3 } ]   ]

[   { LVL   }          [   { 255  } ]   ]
[, {        } = ( dec3 [, {       } ] ) ]
[   { LEVEL }          [   { dec3 } ]   ]

[, { SYS    }         ]
[, {        } = name4 ] ;
[, { SYSTEM }         ]
```

Parameter	Description
NAME	*NAME=name8:* Identifies the filter. No default. The naming convention used for default filters is: FL<*type*>*nnn*, where: – *nnn* is 3 decimal digits – and *type* is NOI, NCC, NAD, ASF, AUT or LOG.
TYPE	Determines the direction of data flow: *TYPE=IN alias INPUT:* Data received by the AF from the AC. *TYPE=OUT alias OUTPUT:* Data received by the AC from the AF. No default. If the type which does not apply to the owner is specified: – either the filter is rejected at generation time – or the filter is ignored at execution time.
AC alias **ADMCOR**	*AC=name8:* Identifies the Administrative Correspondent with which the filter is associated.
AG alias **ADMGROUP**	*AG=name8:* Identifies the Administrative Group with which the filter is associated.
LG alias **ADMLOG**	*LG=name8:* Identifies the set of Logfiles with which the filter is associated.

ADMCOR, ADMGROUP, ADMLOG are mutually exclusive.

LOGIC	Determines the acceptance or rejection of the record: *LOGIC=INCL alias INCLUSIVE:* ***Default*** The filter selects all records whose selection parameters match its criteria. *LOGIC=EXCL alias EXCLUSIVE:* The filter rejects all records whose selection parameters match its criteria. *LOGIC=MAND alias MANDATORY:* The filter rejects all records whose selection parameters do not match its criteria.

ST alias **STATE**

Defines whether the filter is to be considered or ignored:

ST=ENBL: ***Default***
Filter is taken into account.

ST=LOCK: Filter is ignored.

SELECTION CRITERIA

Except for SYSTEM, all criteria are specified as a range between a lower value (first) and an upper value (second). Only one value specified means the lower value. The default upper value is internally supplied. The lower value must always be less than the upper one.

CAUTION:

For performance reasons, from GCOS 7 release V6 onwards, there is no default for the lower value. Missing criteria will be ignored by the network administrative servers.

INCLUSIVE or MANDATORY logic requires at least one criterion. Otherwise the filter is rejected as being inoperable or meaningless.

EXCLUSIVE logic without any criteria specified results in the maximum range of values being internally supplied for each criterion; SYSTEM is not affected.

DOM alias **DOMAIN**

Specifies the administration domain.

DOM=(dec1[,dec1])

Default upper value: 4.
0: DSA or OSI communications
1: Administration of local system
2: Application administration
3: Control of administration
4: Security administration

PWR alias **POWER**

Specifies the administrative information power level.

PWR=(dec1 [,dec1])

Default upper value: 3.
0: Message reporting event
1: Information display command
2: Test and maintenance command
3: Control and modification command

CLASS

Specifies the object class.

CLASS=(dec3 [,dec3]), where 1 <= *dec3* <= 255.

Default upper value: 255.

1: XC cross network monitoring
2: XN cross network transfers
3: SU system startup control
4: PL physical line
5: LL logical line
6: NS network subscription
7: NR network route (path)
8: VC virtual circuit
9: TS transport station
10: MB mailbox
11: SS session
12: LC logical connection
13: CT communications controller
14: DV terminal device
15: AF administrative function
18: FX administrative function connection
19: RS executive (remote subsystem)
EX executive (local OS)
20: SY system
21: TC transport connection
22: WS workstation
23: NC network connection
24: SD session type descriptor
25: CH intersystem channel
26: SN station (terminal)
27: CL terminal cluster
28: LK link connection
29: PC physical connection
30: SC session control
31: CC channel connection
34: FL output filter management (in NAD)
35: SB statistic block
36: NA additional control of NAD itself
38: MU mailbox user
43: AC administrative correspondent
44: SR session route
45: LD logical terminal device
48: OP operator
49: LG log control
51: TL diagnostic application control
53: DF user application control

54: FT file transfer
56: AG administrative group
57: CO correspondent
58: PS physical subscription
59: SG subscription group
60: SX station connection
61: LX logical device connection
62: DX device connection
63: UD user descriptor
64: LN login descriptor
65: CD connection descriptor
66: TX terminal mailbox extension
68: DP device pool
69: WM welcome message
70: TU terminal unit
71: MD terminal model
74: SW software module
75: NU network user
86: AD osi address
87: ET osi layer entity title
88: SP sap address
90: CB cable
91: AI address information
95: AL address list
96: AP application
97: AX application connection
98: UT user of transport
100: VH vehicle (system type specific)
101: UA user agent
102: AS association
105: IS iso session connection
106: ID iso session descriptor
110: RQ dsas request (gcos-8 specific)
111: RB rib - routing information base
112: ML multilink
113: MK multilink connection
114: IK internet activity
115: NK network internet activity
120: FR ftam regime
121: FS ftam file selection regime
122: FE ftam file exchange
(access and transfer regime)
123: FM ftam file management
130: PA printer server agent
140: MA message transfer server agent

141: MQ message queue of ma
142: MM mailbox of ma
143: MI mail item
150: FA document file server agent
151: FF file of document file server agent
152: FD folder of a file of ff
160: DA directory server agent
201 - 255: reserved for user-defined objects

CODE alias **OPCODE**

Defines the action to be taken depending on the object class.

CODE=(dec3 [,dec3]), where 1 <= *dec3* <= 255.

Default upper value: 255.

1 - 49: general commands and events

1: NB report NUMBER of objects
2: LS LIST objects
3: DA DISPLAY common attributes of objects
4: HR report HISTORY (statistics) of objects
GH get HISTORY (statistics) of objects
5: UP UPDATE object state or attributes
6: MP alter MAPPING of objects to other objects
7: CR CREATE object
OP object OPEN report
EX EXECUTE
8: DL DELETE object
CL object CLOSE report
12: GA GET all ATTRIBUTES of an object
14: ER ERROR events report
15: TH THRESHOLD violation report
17: OF OPEN FAILURE event report

50 - 149: class-specific events and error reports

50: TX text broadcast
SU submission
51: SF submission failed
RL reload system
DS dissociation
SW swap
52: DE delivery
LD load system
AS association
RI neighbor requests initialization

53: DF delivery failed
DP dump system
54: RE reception
ST start system
55: TR transmission
56: TF transmission failed
57: DI discard
60: CS completed startup
100: FC file creation
101: FD file deletion
102: FN file rename
103: AC attribute change

150 - 255: implemention specific events and error reports

150: EM executive % use
(per CPU on a multi-CPU machine)
160: DR diverting

LVL alias **LEVEL**

Defines the administrative event importance level.

LVL=(dec3 [,dec3]), where 0 <= *dec3* <= 255.

Default upper value: 255.

0: Response to NUMBER or LIST command
2: Response to ATTRIBUTE or HISTORY reporting commands
4: unimportant errors not usually user-visible
6: answer to modification command
8: impossible answer by NAD and rejected
10: standard events and probe points
12: important events
14: inline test messages to operators
16 and above: important errors
22 and above: operator intervention suggested
24 and above: operator intervention required
26 and above: part of distributed system failed or shutdown
32 and above: fatal errors

SYS alias **SYSTEM**

SYS=name4: Identifies the local or a remote system.

Only the name of the local system must be used if AG is _LDSALOG or AC is LDSALOG.

Default Filters:

1. Attached to **AGs of types: NOI or NCC**.

```
        {NOI}
FL  FL{    }nnn AG=name-8 TYPE=IN LOGIC=INCL DOMAIN=(0,4);
        {NCC}

        {NOI}
FL  FL{    }nnn AG=name-8 TYPE=OU LOGIC=INCL LEVEL=(4,255);
        {NCC}
```

2. Attached to **AGs of types: NAD**.

```
FL  FLNADnnn AG=RLNOI TYPE=IN LOGIC=INCL DOMAIN=(0,4);
```

3. Attached to **AGs of type ASF**.

```
FL  FLASFnnn AG=name-8 TYPE=OU LOGIC=INCL DOM=(0,0)
                       LVL=(12,255);

FL  FLASFnnn AG=name-8 TYPE=OU LOGIC=INCL DOM=(2,3);
```

4. Attached to **AGs of type AUT**.

```
                {RLNAD }
FL  FLAUTnnn AG={      } TYPE=IN LOGIC=INCL DOMAIN=(0,4);
                {name-8}

                {RLNAD }
FL  FLAUTnnn AG={      } TYPE=IN LOGIC=EXCL CODE=(5,8);
                {name-8}

                {RLNAD }
FL  FLAUTnnn AG={      } TYPE=OU LOGIC=INCL DOMAIN=(0,4);
                {name-8}

                {RLNAD }
FL  FLAUTnnn AG={      } TYPE=OU LOGIC=EXCL DOMAIN=(0,4);
                {name-8}
```

5. Attached to **AG _LDSALOG (type NAD)**.

```
FL  FLNADnnn AG=_LDSALOG TYPE=IN LOGIC=INCL DOMAIN=(0,4);
```

6. Attached to **LOG file sets**.

```
                {DSALOG }
FL  FLLOGnnn LG={       } TYPE=OU LOGIC=INCL DOMAIN=(0,4);
                {name-8 }
```

4.7 LG (ADMLOG) Directive

Purpose

This directive applies only to the basic network configuration description. It defines a set of files where AEP records may be logged. These records can be filtered.

Syntax

```
{ LG     }
{        }      NAME = name8
{ ADMLOG }

[  { ST    }   { ENBL } ]
[, {       } = {      } ] ;
[  { STATE }   { LOCK } ]
```

Parameter Description

NAME

Identifies the set of logfiles:

NAME=DSALOG: Defines the mandatory system set of 9 logfiles whose names are: SYS.DSALOG1 through SYS.DSALOG9.
When not specified, is automatically created.

NAME=name8: Defines the optional set of 9 private logfiles whose names are of the format: SYS.<*name8*>1 through SYS.<*name8*>9.

Output filters may be explicitly attached to both sets of logfiles. By default, a filter is attached to each set which allows all UM to be written in the logfiles. See FL directive.

ST alias **STATE**

Defines the initial state of the set of logfiles:

ST=ENBL: ***Default***
Journalizing can start on the set of logfiles when the communications session starts.

ST=LOCK: The set of logfiles awaits operator intervention to be functional.

4.8 SB (STATISTICS_BLOCK) Directive

Purpose

This directive applies only to the basic network configuration description. For network administration purposes, Statistics Block contains the information needed by the SB monitoring mechanism, namely:

- the history report sampling frequency
- the initial state of the SB, locked or enabled
- the detailed criteria identifying the connection objects to be monitored such as class, name, type and state.

During a communications session, the SB monitoring mechanism works only when the server RAEH is active, that is in the USED state, for controlling the emission rate of history reports on connection objects.

Some information within each SB may be modified by the network operator at any time during the communications session, even if the server RAEH is not active, that is in the ENBL state.

Two or more SB directives selecting exactly the same connection object (same class and same object mapping) are declared redundant and signaled by a severity 2 error if their state is explicitly stated as enabled. In this case only one of the directives will be enabled and all other redundant directive(s) will be locked.

For each class, some default SBs may be automatically generated when they do not conflict with those explicitly stated through the SB directives.

Syntax Format 1: Channel Connection

```
{ SB               }
{ STATISTICS       }       NAME = name8
{ STATISTICS_BLOCK }

 {  CL    }
,{        } = { CC | CCONNECT }
 {  CLASS }

[ { SVR | SERVER  } = name8    } ]
[,{                            } ]
[ { LCT | LOCCTLR } = CChexa2 } ]

[ {  HRDELAY       }   { 300  } ]
[,{                } = {      } ]
[ {  HISTORY_DELAY }   { dec6 } ]
```

```
[ {  ST    }   { LOCK } ]
[,{        } = {      } ]
[ {  STATE }   { ENBL } ]

[ {  OBJST    }   { USED } ]
[,{           } = {      } ];
[ {  OBJSTATE }   { ENBL } ]
```

Syntax Format 2: ISO Session

```
{ SB                }
{ STATISTICS        }       NAME = name8
{ STATISTICS_BLOCK }

 {  CL    }
,{        } = { IS | ISOSESS }
 {  CLASS }

[ {  HRDELAY       }   { 300  } ]
[,{                } = {      } ]
[ {  HISTORY_DELAY }   { dec6 } ]

[ {  ST    }   { LOCK } ]
[,{        } = {      } ]
[ {  STATE }   { ENBL } ]

[             { { LOC | LOCAL } alias { INT | INTERNAL  } } ]
[, OBJTYPE = {                                           } ] ;
[             { { RMT | REMOTE } alias { EXT | EXTERNAL } } ]
```

Syntax Format 3 : Logical Connection

```
{ SB                }
{ STATISTICS        }       NAME = name8
{ STATISTICS_BLOCK }

 {  CL    }
,{        } = { LC | LCONNECT }
 {  CLASS }

[ {  HRDELAY        }   { 300  } ]
[,{                 } = {      } ]
[ {  HISTORY_DELAY }   { dec6 } ]

[ {  ST    }   { LOCK } ]
[,{        } = {      } ]
[ {  STATE }   { ENBL } ]

[             { { LOC | LOCAL } alias { INT | INTERNAL }  } ]
[, OBJTYPE = {                                            } ]
[             { { RMT | REMOTE } alias { EXT | EXTERNAL } } ]

[ { LMBX       }   {  name8  } ]
[,{            } = {         } ]
[ { LOCMAILBOX }   {  $name7 } ]

[ { RMBX       }   {  name8  }     {  RSC     }         ]
[,{            } = {         }    ,{          } = name4 ];
[ { RMTMAILBOX }   {  $name7 }     {  RMTSESS }         ]
```

Syntax Format 4: Physical Connection

```
{ SB                }
{ STATISTICS        }       NAME = name8
{ STATISTICS_BLOCK }

{  CL    }
{,       } = { PC | PCONNECT }
{  CLASS }

{  LPL | LOCPLINK }
{,                } = name8
{  RPL | RMTPLINK }

[ {  HRDELAY        }   { 300  } ]
[ {,                } = {      } ]
[ {  HISTORY_DELAY }   { dec6 } ]

[ {  ST    }   { LOCK } ]
[ {,       } = {      } ];
[ {  STATE }   { ENBL } ]
```

Syntax Format 5: DSA Session

```
{ SB               }
{ STATISTICS       }       NAME = name8
{ STATISTICS_BLOCK }

 {  CL    }
,{        } = { SS | DSASESS }
 {  CLASS }

[ {  HRDELAY       }   { 300  } ]
[,{                } = {      } ]
[ {  HISTORY_DELAY }   { dec6 } ]

[ {  ST    }   { LOCK } ]
[,{        } = {      } ]
[ {  STATE }   { ENBL } ]

[             { { LOC | LOCAL } alias { INT | INTERNAL }  } ]
[, OBJTYPE = {                                            } ] ;
[             { { RMT | REMOTE } alias { EXT | EXTERNAL } } ]
```

Syntax Format 6: TNS Transport Connection

```
{ SB               }
{ STATISTICS       }       NAME = name8
{ STATISTICS_BLOCK }

 {  CL    }
,{        } = { TC | TCONNECT }
 {  CLASS }

[ {  RTS          }         ]
[,{               } = name8 ]
[ {  RMTTRANSPORT }         ]

[ {  HRDELAY       }   { 300  } ]
[,{                } = {      } ]
[ {  HISTORY_DELAY }   { dec6 } ]

[ {  ST    }   { LOCK } ]
[,{        } = {      } ]
[ {  STATE }   { ENBL } ]

[ {  OBJST     } = { USED } ]
[,{            } = {      } ];
[ {  OBJSTATE  } = { ENBL } ]
```

Syntax Format 7: Internet Network Activity

```
{ SB              }
{ STATISTICS      }      NAME = name8
{ STATISTICS_BLOCK }

 {  CL    }
,{        } = { NK | INETWORK }
 {  CLASS }

[ {  RTS          }         ]
[,{               } = name8 ]
[ {  RMTTRANSPORT }         ]

[ {  ST    }   { LOCK } ]
[,{        } = {      } ] ;
[ {  STATE }   { ENBL } ]
```

Parameter Description

NAME	*NAME=name8:* Identifies the Statistics Block. No default.
CL alias **CLASS**	**Applicable to all Formats.** Defines the class of connection objects monitored through the current statistics block, each class having a different format.
ST alias **STATE**	**Applicable to all Formats.** Defines the initial state of the statistics block at the start of the communications session. This state can be later changed by the MDNET SB network command. *ST=LOCK:* ***Default*** The SB is not operational and does not send the history report messages concerning its connection objects. *ST=ENBL:* The SB is operational and starts sending history reports on its connection objects, as soon as the RAEH server is started.
HRDELAY alias **HISTORY_DELAY**	**Applicable to all Formats.** Defines the minimum time interval, expressed in seconds, between two successive message emissions of history report. This value can be changed through the INTERVAL parameter of the MDNET SB network command. 60 <= *hrdelay* <= 999999. ***Default:*** 5 minutes (300 seconds).

OBJST alias **OBJSTATE**

Applicable only to Formats 1 and 6. The connection objects are to be monitored only if they are in the specified state. When omitted, these objects are monitored whatever their state.

OBJST=USED: ***Default***
The connection object is to be monitored only when it is in the *used* state.

OBJST=ENBL: The connection object is to be monitored only when it is in the *enabled* state.

OBJTYPE

Applicable only to Formats 2, 3 and 5.

OBJTYPE=LOC alias LOCAL ***or***
OBJTYPE=INT alias INTERNAL:
Only the local connection objects are to be monitored.

OBJTYPE=REM alias REMOTE ***or***
OBJTYPE=EXT alias EXTERNAL:
Only the remote connection objects are to be monitored.

SVR alias **SERVER**

Applicable only to Format 1 (Channel Connection).

SVR=name8: Only channel connections mapped to the server are to be monitored.
SVR is mutually exclusive with LCT, one of which is mandatory.

LCT alias **LOCCTLR**

Applicable only to Format 1 (Channel Connection) and to controllers of type PSI.

LCT=CChexa2: Only channel connections mapped to the local PSI controller are to be monitored.
LCT is exclusive mutually with SVR, one of which is mandatory.

LMBX alias **LOCMAILBOX**

Applicable only to Format 3 (Logical Connection).
LMBX=name8 or $name7: Only logical connections mapped onto the local mailbox are to be monitored.

- When both LMBX and RSC-RMBX are specified, only the logical connections mapped onto both local and remote mailboxes are monitored.
- When neither LMBX nor RMBX is specified, all the logical connections are to be monitored.

RMBX alias **RMTMAILBOX**

Applicable only to Format 3 (Logical Connection).

RMBX=name8 or $name7: Only logical connections mapped onto the remote session control and mailbox are to be monitored. *RMBX must be specified with RSC.*

- When both LMBX and RSC-RMBX are specified, only the logical connections mapped onto both local and remote mailboxes are monitored.
- When neither LMBX nor RMBX is specified, all the logical connections are to be monitored.

RSC alias **RMTSESS**

Applicable only to Format 3 (Logical Connection).

RSC=name4: Only logical connections mapped onto the remote session control and mailbox are to be monitored. *RSC must be specified with RMBX.*

- When both LMBX and RSC-RMBX are specified, only the logical connections mapped onto both local and remote mailboxes are monitored.
- When neither LMBX nor RMBX is specified, all the logical connections are to be monitored.

LPL alias **LOCPLINK**
RPL alias **RMTPLINK**

Applicable only to and mandatory for Format 4 (Physical Connection).

LPL=name8 ***and***
RPL=name8: Only physical connections mapped onto the specified local or remote physical link are to be monitored.

RTS alias **RMTTRANSPORT**

Applicable only to Format 6 (Transport Connection). RTS is optional and is meaningful only when TNS is specified.

RTS=name8: Only transport connections mapped onto the remote transport station are to be monitored. If omitted, all TNS transport connections accessing the local transport station are monitored.

Default Statistics Blocks

The following SB directives correspond, for each class, to the default statistics blocks. If at least one SB is explicitly declared for a given class, no default SB is internally generated for this class. Default SBs are always created in the default LOCK state so that they may be later *enabled* by the network operator.

1. Channel Connection (one per server or local controller):

```
SB     _SBCCdec2, CLASS = CC, SVR = name8;
```

2. ISO Session (single one):

```
SB     _SBIS0   , CLASS = IS;
```

3. Logical Connection (single one):

```
SB     _SBLC0   , CLASS = LC;
```

4. Physical Connection (one per local or remote physical link):

```
SB     _SBPCdec2, CLASS = PC, LPL = name8;
SB     _SBPCdec2, CLASS = PC, RPL = name8;
```

5. DSA Session (single one):

```
SB     _SBSS0    CLASS = SS;
```

6. TNS Transport Connection (one per remote transport station):

```
SB     _SBTCdec2, CLASS = TC;
```

7. Internet Network Activity:

```
SB     _SBNK0, CLASS = NK;
```

4.9 Network Administration Example

I- Schematic Description

The following example shows:

- a limit to 10 (otherwise the default should be 32) of the maximum number of sessions which can be established concurrently by all the ACs grouped under the AGs: RLNAD, RDSALOG, RLNOI and RMUX
- a limit to a single session for the AG named RASF which specifies INITIATOR=AF
- a set of three static ACs for this AG:
  - AC named LOGF declared primary which represents a remote ASF function located on the system BPA0 with mailbox $LOGFILE
  - Two similar ACs named LOGFBK2 and LOGFBK3 both declared backup and located respectively on the remote systems BP3C and BP10
- an output filter attached to the primary AC which allows all classes of UMs to be logged on system BPA0 (the default filtering attached to the AG RASF being more restrictive: LEVEL=12(lower),255(upper)). Explicitly declaring this filter allows later dynamic modifications by operator commands.

Directives describing the corresponding basic network configuration are omitted.

II- Network Administration Directives

```
NETWORK  BPA1;

COMM '==================================================';
COMM '=    BASIC NETWORK CONFIGURATION DESCRIPTION    =';
COMM '=             IS    NOT    SHOWN                =';
COMM '==================================================';

COMM '==================================================';
COMM '=       ADMINISTRATIVE OBJECTS GENERATION       =';
COMM '==================================================';
AG    RLNAD,    MAXSESS = 10;
AG    RDSALOG,  MAXSESS = 10;
AG    RLNOI,    MAXSESS = 10;
AG    RMUX,     MAXSESS = 10;
AG    RASF      MAXSESS = 1, TYPE = ASF, AF= LNAD, MAXEV=500,
                INIT = AF, RETRYDELAY = 1;
AC    LOGF      TYPE = ASF AG = RASF SC = BPA0, MBX=$LOGFILE;
AC    LOGFBK2   TYPE = ASF AG = RASF SC = BP3C, MBX=$LOGFILE,
                BACKUP = 1;
AC    LOGFBK3   TYPE = ASF AG = RASF SC = BP10, MBX=$LOGFILE,
                BACKUP = 1;
FL    FLACLOGF  AC = LOGF TYPE= OUT LOGIC=INCL CLASS=(1,255);

COMM '==================================================';
COMM '=         STATISTICS OBJECTS GENERATION         =';
COMM '==================================================';
SB    SBCCCC01 CLASS=CC LCT=CC01 STATE=ENBL;
SB    SBCCSV10 CLASS=CC SVR=SVR_BP10 HRDELAY=120;
SB    SBLCRMB1 CLASS=LC RMBX=MBX_APP1 OBJTYPE=LOCAL;
SB    SBPCEA11 CLASS=PC LPL=LPL_EA11;
SB    SBTCBP12 CLASS=TC RTS=BP12 OBJST=USED ST=ENBL;

ENDNETWORK BPA1;
```

❑

5. Communications Service of Correspondents

5.1 PPC and XCP2 Protocol

XCP2 is the protocol used by PPC (Program-to-Program Communication) to provide services so that programs can be:

- local to the same GCOS 7 system
- distributed over different GCOS 7 systems through a DSA network using XCP2 protocol
- distributed over GCOS 7 systems and foreign (IBM) systems supporting LU6.2 protocol in a mixed DSA-SNA network, an OSF function ensuring the conversion between XCP2 and LU6.2.

Interprogram communication is accessible to a program via a TP_Node (Transaction Processing Node) which handles the resources required.

In order to function, XCP2 needs to retrieve the following objects from the directory configuration (correspondent dictionary):

- XCP2COR (mandatory)
- XCP2WKS (optional)
- XCP2POOL (optional).

Since TDS is the only application using XCP2 protocol, a TDSWKS must be configured.

Between two *cold starts* of the TP-Node, XCP2 cannot take into account the modifications due to an incremental generation of the directory configuration:

- XCP2COR: PPC retrieves in the directory configuration the address of the partner correspondent at each first connection
- XCP2WKS: PPC uses the same description of the XCP2WKS retrieved at the opening of the TP_Node
- XCP2POOL: PPC uses the pool description retrieved in the directory configuration at the opening of the pool.

5.1.1 TP_Node or XCP2WKS (XCP2 Workstation)

The TP_Node is the entity which handles for a given application, the resources necessary for inter-program communication such as session pools, conversations, and the control structures, queues and processes involved in transaction processing. An application wanting to use PPC services must first open a TP_Node. For example, if three applications want to use PPC services, each must first open its own TP_Node.

The XCP2WKS (XCP2 Workstation) is the representation of a TP_Node specified in the NETGEN correspondent configuration description. So term *TP_Node* used in the PPC environment and the term *XCP2WKS* in the NETGEN description are *identical*. When a PPC node is opened, PPC gets the description of the workstation either from its caller for example TDS or, if the caller does not supply it, from the directory configuration through the XCP2WKS directive.

When the application supplies to PPC a workstation description, it may:

- either directly retrieve it from the directory configuration and possibly modify part of it before calling PPC
- or entirely build it up without accessing the directory configuration.

PPC uses this description between two *cold starts* of the TP-Node corresponding to two *cold starts* of the TDS application. The application may modify some parameters of the workstation description directly in the PPC structures.

The following workstation attributes may be configured in the directory to specify limits in the size or the number of occurrences for some PPC objects to allow PPC to tailor its own data structures:

- MAXPROCESS represents the maximum number of physical application processes that PPC will use.
- MAXSESS specifies two values, the first being the maximum number of sessions supported by the VCAM mailbox attached to the TP_Node, the second being the maximum number of sessions supported by the TP_Node.
- MAXTRANSAC is the maximum number of transactions that the TP_Node can support.
- MAXCOR is the maximum number of correspondents that PPC can support.
- MAXPOOL represents the maximum number of session pools supported by the TP_Node. For a *single* correspondent, PPC reserves space in its own structures for at least one pool.
- MAXCONV represents the maximum number of conversations for a transaction program. Even in the case where the conversation occurs between a TP-Node and only a *single* correspondent, MAXCONV can be greater than 1.

- CONVBUF_MAXSZ and SESSBUF_MAXSZ being the two types of buffers for receiving and sending data either to and from the conversation, or to and from the session involving the partner correspondent. The size of the conversation buffer must be at least equal to the size of the session buffer.
- PREFIX_UNIQUE_NAME and UNIQUE_NAME uniquely identify the TP_Node in the network. If the TP_Node is connected to an SNA network, the prefix which identifies the SNA network and the unique name which identifies the TP_Node, together make up the fully qualified LU-name.

5.1.2 XCP2COR (XCP2 Correspondent)

A TP_Node establishes sessions with a *partner* TP_Node. The *partner* TP_Node is identified by the local system as a correspondent whose name is associated with its network address by which it is accessed.

All the correspondents which are likely to be used in XCP2 connections must be declared in the NETGEN directory configuration description.

Two kinds of connection can be used between the *local* TP_Node and its *partner* TP_Node:

- *single* connection in which only one session can be established between the two TP_Nodes, in which case the TP_Node is itself *single*
- *parallel* connection in which more than one session can be established between the two TP_Nodes, in which case the TP_Node is itself *parallel*.

Partner correspondents must be declared in the directory configuration using the XCP2COR directive. The parameters PRIMARY, WINNER, SYNC_LEVEL and the security parameters are meaningful only in the case of single session. Otherwise, the information is taken from the pool description.

The directory configuration must contain one additional correspondent, PPC's own TP-node (XCP2WKS) which supplies to PPC its address.

If the partner correspondent is local to the same GCOS 7 system, its network address (session + mailbox) can be replaced in the directive by the name of the XCP2WKS describing the correspondent.

A correspondent or an XCP2 pool may have two levels of synchronization, *confirm* and *syncpoint*. A conversation however, may also have a third level of synchronization, *none*. See SYNC_LEVEL.

At the first connection with a partner correspondent, PPC retrieves in the directory configuration the address of the correspondent.

5.1.3 XCP2POOL (XCP2 Pool of Sessions)

Sessions between TP-Nodes are described in one or several session pools. A session pool is identified by a correspondent-name and a pool-name. The session pool is managed through operator commands and primitives of the programmatic interface.

Before being allocated to a TP, session pools must first be opened by either the local or a remote TP_Node:

- when the TP_Node is *single*, opening the pools is optional and the options of the NETGEN correspondent are then taken into account
- when TP_Nodes are *parallel*, opening the pools is mandatory before they can be used.

Several pools may be attached to the same correspondent, each pool having different functional characteristics such as SYNC_LEVEL. The pool description in the directory configuration is optional.

The pool description is searched for in the directory configuration by the caller when the pool is opened. If the pool is not configured, the XCP2 operator must supply the whole pool description when opening the pool.

The description retrieved in the directory configuration is valid between two pool openings during the same PPC session, corresponding to two *cold starts* of the TP-Node or the TDS application.

The security parameters defined for the XCP2WKS may be redefined at pool level.

5.1.4 Winner or Loser Session

A session is a resource shared between two TP-Nodes. As contention for using the session may arise between two partner correspondents, one of the TP_Nodes must be a *winner* and the other a *loser*. This decision is taken at session connection time.

A *winner* TP_Node does not require authorization to use the session. A *loser* TP_Node must always request authorization.

5.1.5 TP (Transaction Program)

A TP (Transaction Program) is a program using PPC services. For PPC, the *TP-entity* consists of only attached PPC structures. PPC provides services for distributed cooperative programs using at least two TPs, one attached to the *local* TP_Node and the other to the *partner* TP_node.

5.1.6 Conversation

While the session is the link between two TP_Nodes, the conversation is the link between two TPs. A conversation uses the session as a resource. A session remains allocated to a conversation until it is deallocated making it available again for another conversation.

5.1.7 SYNC_LEVEL (Synchronization Level)

Three levels of synchronization are defined for the conversation:

- **None**: no synchronization service can be used on the conversation.
- **Confirm**: a program can request its partner through the *confirm* service, to send a confirmation. The partner does so using the *confirmed* service. This allows the two partners to implement their own synchronization mechanism.
- **Syncpoint**: in addition to *confirm* and *confirmed* services, *syncpoint* and *backout* are also supported:
  - *syncpoint* which defines a *commitment point* for the transaction
  - *backout* which resets resources to their status at the last *successful* syncpoint.

Only two levels of synchronization are defined at pool and correspondent level, *syncpoint* and *confirm*. A conversation synchronization level of *none* is the lowest level and can be supported whatever the synchronization level of pool and correspondent.

5.1.8 Security

Security checking concerns conversations and sessions for either the inward or the outward connections.

Security parameters are defined at different levels, XCP2 workstations and pools of sessions. Conversation security parameters may be redefined by a TP when a conversation is established.

5.2 Correspondent Directives

General Syntax Rules

Within the source corresponding to the directory configuration the directives describing the service of correspondent objects may be entered in any order between the DIR and EDIR directives, these being the first and last directives. To avoid confusion, they should not be merged with the directives describing subdomains and remote systems.

OPEN 7 ENVIRONMENT

Where the DPS 7000 works in the OPEN 7 environment and communicates using only the TCP/IP protocols, only the local system need be supplied in the basic network configuration description. The directory configuration description then will only contain correspondent objects related to the local system and local applications.

The following schema shows the directive sequence for a typical directory configuration (correspondent dictionary). For the description and syntax of subdomain and remote system directives, see Section 3.

```
DIRECTORY

[ subdomains directives     ]

[ remote systems directives ]
   .
   .
   .
[ correspondents directives ]
   .
   .
   .
[ ENDDIRECTORY ]
```

NOTES:

1. For all directives, parameters can be specified in any order. All arguments are introduced by their corresponding keywords. NAME is the only exception being the first positional and therefore can be omitted.
2. In any directive when the argument is boolean (0 or 1), the keyword on its own is equivalent to specifying 1 as its argument, for example:
   PRIMARY is equivalent to PRIMARY=1,
   NPRIMARY is equivalent to PRIMARY=0.

List of Correspondent Directives

In this Section, the NETGEN directives are described in the alphabetical order of their abbreviations. The list below puts the same directives in alphabetical order of their expanded acronyms, giving also the corresponding abbreviation, configuration description destination (Dc for service of correspondents) and paragraph number.

Abbr	Expanded	Dest	Par.	Abbr	Expanded	Dest	Par.
TDS	TDSWKS	Dc	5.3	X2	XCP2WKS	Dc	5.6
TMC	TMCOR	Dc	5.4	X2C	XCP2COR	Dc	5.7
X1C	XCP1COR	Dc	5.5	X2P	XCP2POOL	Dc	5.8

5.3 TDS (TDSWKS) Directive

Purpose

This directive applies only to the directory configuration description. It defines the communications parameters of a TDS application.

Syntax

```
{ TDS    }
{        }        NAME = name4
{ TDSWKS }

[ { X1MBX   }         ]
[,{         } = name8 ]
[ { XCP1MBX }         ]

[ { TMSC   }   { 100  } ]
[,{        } = {      } ]
[ { TMSESS }   { dec4 } ]

[ { X1SC     }   { 0    } ]
[,{          } = {      } ]
[ { XCP1SESS }   { dec4 } ]

[ { MAST   }   { SUBM | SUBMITTER } ]
[,{        } = { X1 | XCP1        } ]
[ { MASTER }   { TM               } ]

[ { MASTMBX        }         ]
[,{ MASTERMBX      } = name8 ]
[ { MASTER_MAILBOX }         ]

[ { X2      }         ]
[,{         } = name8 ] ;
[ { XCP2WKS }         ]
```

Parameter Description

NAME	Identifies of the TDS application. It is mandatory and must be unique and different from XCP2WKS. *NAME=name4:* The name of the mailbox to which the terminals are connected is the application name.

X1MBX alias **XCP1MBX** *X1MBX=name8:* XCP1 mailbox name which must be unique and different from all other mailboxes.

***Default**:* XCP1*name*, where *name* is the TDSWKS name.

TMSC alias **TMSESS** *TMSC=dec4:* Maximum number of sessions that connect terminals to the TDS application.

Default: 100.

X1SC alias **XCP1SESS** *X1SC=dec4:* Maximum number of XCP1 sessions between the current TDS application and the set of applications to which it is connected.

***Default**:* 0 **-** the TDS application cannot use XCP1 protocol.

MAST alias **MASTER** Defines the type of dialog with the master operator:

MAST=SUBM alias SUBMITTER: The IOF user that starts the TDS is the master operator.

MAST=X1 alias XCP1: The dialog with the master operator is supported by an XCP1 session.

MAST=TM: The dialog with the master operator is supported on a DSA terminal session.

MASTMBX alias **MASTERMBX** alias **MASTER_MAILBOX**
Not applicable when MASTER=SUBMITTER.

MASTMBX=name8: Identifies the mailbox to which the master sessions connect. It must be unique and different from all other maiboxes.

***Default**:* MAST*name*, where *name* is the TDSWKS name.

X2 alias **XCP2WKS** *X2=name8:* Identifies the XCP2WKS which must be declared in an X2 directive. When specified the TDS application can use XCP2 services. No default.

5.4 TMC (TMCOR) Directive

Purpose

This directive applies only to the directory configuration description.

A terminal manager correspondent represents a user logged from a terminal on a transactional application. This directive defines its attributes and mainly its network address.

Syntax

```
{ TMC   }
{       }   [ NAME = name12 ]
{ TMCOR }

[ { SC   }          ]
[,{      } = name4  ]
[ { SESS }          ]

  { MBX     }
 ,{         } = name8
  { MAILBOX }

[ { BACK   }              [ { FIRST        }   { 1 } ]]
[,{        } = name12    [,{ FIRSTBACK    } = {   } ]] ;
[ { BACKUP }              [ { FIRST_BACKUP }   { 0 } ]]
```

Parameter Description

NAME	*NAME=name12:* Identifies the correspondent and must be unique. ***Default**: NAME* of a X1C or X2C directive previously declared.
SC alias **SESS**	*SC=name4:* Identifies the Session Control of the target system and must be declared by a LSC or RSC directive in either the basic network or the directory configurations. ***Default**: SESS* of a XCP1COR or XCP2COR or TMCOR directive previously declared.

MBX alias **MAILBOX** *MBX=name8:* Identifies the Mailbox of the target terminal. This parameter is mandatory. No default.

CORRESPONDENT ADDRESS

Defined by parameters: SESS + MAILBOX.

Two TM correspondents must not have the same address.

BACK alias **BACKUP** *BACK=name12:* Identifies the backup correspondent and must be declared by a TMCOR directive with opposite FIRST value.

FIRST alias **FIRSTBACK** alias **FIRST_BACKUP**

Indicates for a pair of cross declared correspondent backups, which one is privileged. Both backups must not have the same FIRST value. ***FIRST is meaningless when BACKUP is not specified.***

FIRST=1: ***Default***
The current correspondent is privileged.
FIRST=0: The current correspondent is not privileged.

5.5 X1C (XCP1COR) Directive

Purpose

This directive applies only to the directory configuration description. It represents a remote application which can cooperate using the XCP1 protocol. This directive defines its attributes and mainly its network address and its profile.

Syntax

```
{ X1C     }
{         }          [ NAME = name12 ]
{ XCP1COR }

 { [ { SC   }        ] { MBX     }       [ { EXT       }       ] }
 { [ {      }=name12 ],{         }=name8 [,{           }=name4 ] }
 { [ { SESS }        ] { MAILBOX }       [ { EXTENSION }       ] }
,{                                                              }
 { { TDS    }        }                                          }
 { {        } = name4 }                                         }
 { { TDSWKS }        }                                          }

[ { PRIM    }   { 1 } ]
[,{         } = {   } ]
[ { PRIMARY }   { 0 } ]

[ { ACT    }   { 1 } ]
[,{        } = {   } ]
[ { ACTIVE }   { 0 } ]

[ { INITW    }   { 0 } ]
[,{          } = {   } ]
[ { INITWORK }   { 1 } ]

[ { BACK   }          [ { FIRST        }   { 1 } ]]
[,{        } = name12  [,{ FIRSTBACK    } = {   } ]] ;
[ { BACKUP }          [ { FIRST_BACKUP }   { 0 } ]]
```

Parameter Description

NAME *NAME=name12:* Identifies the correspondent and must be unique.
Default*: NAME* of a TMC or X2C directive previously declared.

SC alias **SESS**	*SC=name12:* Identifies the Session Control of the target system and must be declared by an RSC directive either in the basic network or the directory configuration. ***Default****: SESS* of X1C or X2C or TMC directive previously declared. *SESS is mutually exclusive with TDSWKS.*
MBX alias **MAILBOX**	*MBX=name8:* Identifies the Mailbox of the target application. No default. *MBX is mutually exclusive with TDSWKS and is* ***mandatory*** *when TDSWKS is omitted.*
EXT alias **EXTENSION**	*EXT=name4:* Identifies the Mailbox extension of the target application. *EXT is mutually exclusive with TDSWKS.*

CORRESPONDENT ADDRESS

Defined by parameters: SESS + MAILBOX [+ EXTENSION].

Two XCP1 correspondents cannot have the same address. The correspondent address is mutually exclusive with the TDS parameter. If TDSWKS is specified, the following implicit XCP1 correspondent address is generated:

<local SESS name> + <MAILBOX of the TDSWKS>.

If TDSWKS is not specified, a correspondent address ***must*** be specified.

TDS alias **TDSWKS**	*TDS=name4:* Identifies the TDS application, which can be accessed through XCP1 protocol and must be declared by a TDSWKS directive.
PRIM alias **PRIMARY**	Determines whether the remote or backend correspondent initiates XCP1 recovery: *PRIM=1:* ***Default*** **-** primary *PRIM=0:* secondary.
ACT alias **ACTIVE**	Determines how the transaction is initiated onto the remote application: *ACTIVE=1:* ***Default*** **-** ACTIVE *ACTIVE=0:* passive.

INITW alias **INITWORK** Meaningful for an ACTIVE correspondent.

INITW=0: ***Default***
A *passive* correspondent cannot initiate a transaction towards an ACTIVE correspondent.

INITW=1: Allows a *passive* correspondent to initiate a transaction towards an ACTIVE correspondent.

CORRESPONDENT PROFILE

Defined by parameters: PRIMARY + ACTIVE + INITWORK.

BACK alias **BACKUP** **Meaningless when TDS-HA environment is not used**. *BACK=name12:* Identifies the backup correspondent which must be declared by another XCP1COR directive having:

- the same correspondent profile
- the opposite FIRST attribute.

FIRST alias FIRSTBACK alias FIRST_BACKUP

Meaningless when BACKUP is not specified. For a pair of cross declared backup correspondents it indicates:

FIRST=1: first backup is privileged
FIRST=0: not first backup.

5.6 X2 (XCP2WKS) Directive

Purpose

This directive applies only to the directory configuration description. The XCP2 workstation known as TP-Node, is the main entity which handles for a given local application all the resources necessary for interprogram communication such as session pools and conversations, and control structures, queues and processes required for transaction program resources.

To use XCP2 services, an application has to open its XCP2 workstation. It is the network configuration image of a transactional application using XCP2 protocol.

This directive defines the configuration attributes of an XCP2 workstation. They can be used as default parameters when opening the workstation.

An XCP2 correspondent associated with this XCP2 workstation must be declared using the X2C directive.

Security Parameters

CONV_USERID, CONV_CHECK, CONV_VERIFIED_ACCEPTED apply only for a remote XCP2 application. These parameters define how security checks, if applied, are to be performed when starting up a remote transaction. If these checks are successful, the remote transaction is eligible for startup.

See paragraph 5.9 "Conversation Security Mechanism".

Syntax

```
{ X2      }                           [ { UNAME       }        ]
{         }          NAME=name8       [,{             }=name8 ]
{ XCP2WKS }                           [ { UNIQUE_NAME }        ]

[ { MBX     }       ]                 [ { MAXPCS     } { 10   } ]
[,{         }=name8 ]                 [,{            }={      } ]
[ { MAILBOX }       ]                 [ { MAXPROCESS } { dec5 } ]

[ { MAXSC   }     { 10   } [ { 10   } ]   ]
[,{         } = ( {      } [,{      } ] ) ]
[ { MAXSESS }     { dec4 } [ { dec4 } ]   ]

[ { MAXTX      } { 10   } ]           [ { MAXC   } { 10   } ]
[,{            }={      } ]           [,{        }={      } ]
[ { MAXTRANSAC } { dec4 } ]           [ { MAXCOR } { dec4 } ]
```

```
[          { 10   } ]                       [            { 1    } ]
[,MAXPOOL={       } ]                       [,MAXCONV = {        } ]
[          { dec4 } ]                       [            { dec3 } ]

[ { CONVBUFSZ     }   { 4096 } ]
[,{ CONV_BUFSIZE  } = {      } ]
[ { CONVBUF_MAXSZ }   { dec5 } ]

[ { SCBUFSZ       }     { 4096 } [   { 4096 } ]  ]
[,{ SESS_BUFSIZE  } = ( {      } [, {      } ]) ]
[ { SESSBUF_MAXSZ }     { dec5 } [   { dec5 } ]  ]

[ { SVR    } { 1 } ]                        [ { SYNCPT    } { 0 } ]
[,{        }={   } ]                        [,{           }={   } ]
[ { SERVER } { 0 } ]                        [ { SYNCPOINT } { 1 } ]

[ { CONV_USER   } { OPTIONAL  } ]           [ { CONVCK     } { 1 } ]
[,{             }={ MANDATORY } ]           [,{            }={   } ]
[ { CONV_USERID } { NONE      } ]           [ { CONV_CHECK } { 0 } ]

[ { CONV_ACCEPT            } { 0 } ]        [ { LOG     }          ]
[,{                        }={   } ]        [,{ LOGNAME }=name12 } ]
[ { CONV_VERIFIED_ACCEPTED } { 1 } ]        [ { LOGFILE }          ]

[ { PFX_UNAME          }   { BULLNET } ]
[,{                    } = {         } ] ;
[ { PREFIX_UNIQUE_NAME }   { name8   } ]
```

Parameter Description

NAME — *NAME=name8:* ***Mandatory*** and unique name identifying the XCP2 workstation and must be different from all other workstations and mailboxes.

UNAME alias **UNIQUE_NAME** — *UNAME=name8:* Identifies the current XCP2 workstation as a correspondent among all the XCP2 TP-Nodes of the whole network and is used when contentions occur when sessions are established. ***Default****:* the current XCP2WKS name.

MBX alias **MAILBOX** — *MBX=name8:* Identifies the mailbox attached to the XCP2 workstation.

Default*:* the current XCP2WKS name.

MAXPCS alias **MAXPROCESS**
Defines the maximum number of application processes which can execute the XCP2 services.

1 <= *maxprocess* <= 32767. ***Default:*** 10.

MAXSC alias **MAXSESS** MAXSESS=(dec-4 [,dec-4]): where

- first value represents the number of sessions which can connect to the XCP2 workstation. ***Default:*** 10.
- second value represents the maximum number of sessions supported by the XCP2 workstation. ***Default:*** first value.
- 0 < *1st_value* <= *2nd_value* <= 5000.

MAXTX alias **MAXTRANSAC**
Defines the maximum number of transactions which can be supported by the XCP2 workstation.

0 < *maxtx* <= 1000. ***Default:*** 10.

MAXC alias **MAXCOR** Defines the maximum number of partner correspondents which can be supported by the XCP2 workstation.

0 < *maxcor* <= 5000. ***Default:*** 10.

MAXPOOL Defines the maximum number of pools which can be supported by the XCP2 workstation.

0 < *maxpool* <= 1000. ***Default:*** 10.

MAXCONV Defines the maximum number of conversations which can be allocated to a transaction program.

0 < *maxconv* <= 100. ***Default:*** 1.

CONVBUFSZ alias **CONV_BUFSIZE** alias **CONVBUF_MAXSZ**
Defines the maximum size of conversation buffer.

4096 <= *convbufsz* <= 64512 (63k bytes). ***Default:*** 4096.

SCBUFSZ alias **SESS_BUFSIZE** alias **SESSBUF_MAXSZ**

SCBUFSZ= (dec-5 [,dec-5]): where

- first value represents the maximum size of session buffer for sending data throughout the network. ***Default:*** 4096.
- second value represents the maximum size of session buffer for receiving data from the network. ***Default:*** first value.
- 4096 <= *dec-5* <= 32512.

SVR alias **SERVER**

Determines if transactions or requests from partner correspondents to use the pool are accepted by the XCP2 workstation:
SVR=1: ***Default***
SVR=0: Not accepted.

SYNCPT alias **SYNCPOINT**

Determines if *syncpoint* services are supported.
SYNCPT=1: syncpoint services are supported by the XCP2 workstation
SYNCPT=0: ***Default*** **-** *syncpoint* services are not supported.

CONV_USER alias **CONV_USERID**

Conversation security parameter, see 5.9. Defines the remote security procedure:

CONV_USER=MANDATORY: The XCP2 application must perform conversation security checks depending on the setting of the *already-verified* flag in the incoming request:

- if OFF: PASSWORD must be provided and the request is then checked according to the CONVCK value; otherwise the request is rejected.
- if ON: the request is either rejected if CONV_ACCEPT=0 or checked according to the CONVCK value.

CONV_USER=NONE: No checks; the conversation is always accepted.

CONV_USER=OPTIONAL: ***Default***, Result depends on whether USER is specified in incoming request:

- if specified, acts as CONV_USER=MANDATORY
- if omitted, acts as CONV_USER=NONE.

CONV_ACCEPT alias **CONV_VERIFIED_ACCEPTED**

***Conversation security parameter, see 5.9*. Not applicable with CONV_USER=NONE.**

Specifies whether an incoming request with an *already-verified* flag set to ON is to be accepted.

CONV_ACCEPT=0: ***Default* -** Rejected

CONV_ACCEPT=1: Accepted depending on the checks based on values of the CONVCK.

CONVCK alias **CONV_CHECK**

***Conversation security parameter, see 5.9*. Not applicable with CONV_USER=NONE.**
Checks that the transaction can be invoked on behalf of the PROJECT.

CONVCK=1: ***Default***
The PROJECT must have in its site CATALOG this XCP2 application defined by MAILBOX of the associated XCP2WKS directive, in its application list and has the proper authority code to access it.

CONVCK=0: No project check.

LOG alias **LOGFILE** alias **LOGNAME**

Applicable only if SYNCPOINT=1.

LOG=name12: Identifies the logfile to be used for recovery.
Default*: xcp2wks-name*_LOG, where *xcp2wks-name* is the name of the current XCP2WKS.

PFX_UNAME alias **PREFIX_UNIQUE_NAME**

PFX_UNAME: Represents the network identifier when used with a SNA network.
Default*:* BULLNET

5.7 X2C (XCP2COR) Directive

Purpose

This directive applies only to the directory configuration description. An XCP2 correspondent is the local representation

- either of a local XCP2 workstation declared by a XCP2WKS directive
- or of a remote XCP2 workstation acting as partner with the local one.

This directive defines the correspondent attributes according to XCP2 protocol and mainly its network address and its profile.

Syntax

```
{ X2C     }
{         }            [ NAME = name12 ]
{ XCP2COR }

 { [ { SC   }         ]  { MBX     }         }
 { [,{      } = name4 ] ,{         } = name8 }
 { [ { SESS }         ]  { MAILBOX }         }
,{                                           }
 { { X2      }                               }
 { {         } = name8                       }
 { { XCP2WKS }                               }

[          { 1 } ]                   [ { PRIM    } { 1 } ]
[,PARALLEL={   } ]                   [,{         }={   } ]
[          { 0 } ]                   [ { PRIMARY } { 0 } ]

[ { SYNCLVL    } { CONFIRM   } ]     [ { WIN    } { 1 } ]
[,{            }={           } ]     [,{        }={   } ]
[ { SYNC_LEVEL } { SYNCPOINT } ]     [ { WINNER } { 0 } ]

[ { USERCTL        }   { OPTIONAL  } ]
[,{                } = { MANDATORY } ]
[ { USERID_CONTROL }   { NONE      } ]

[ { PROJCTL         }   { 1 } ]
[,{                 } = {   } ]
[ { PROJECT_CONTROL }   { 0 } ]

[ { BACK   }                  [ { FIRST        }   { 1 } ]]
[,{        } = name12         [,{ FIRSTBACK    } = {   } ]] ;
[ { BACKUP }                  [ { FIRST_BACKUP }   { 0 } ]]
```

Parameter	Description
NAME	*NAME=name12:* Identifies the correspondent and must be unique. ***Default****:* the name of a TMCOR or XCP1COR directive previously declared.

CORRESPONDENT ADDRESS

Defined by parameters: SESS + MAILBOX.

It is mandatory when either X2C is remote or X2C is local and is not declared with the XCP2WKS parameter.

Two XCP2 correspondents cannot have the same address.

The XCP2WKS parameter generates the implicit correspondent address: *<local SESS name> + <MAILBOX of the XCP2WKS>*

SC alias **SESS**	*SC=name4*: Identifies the Session Control of the target system and must be declared by an RSC directive in either the basic network or the directory configuration. *SESS is mutually exclusive with XCP2WKS.* ***Default****:* the session control name of a TMC or X1C or X2C directive previously declared.
MBX alias **MAILBOX**	*MBX=name8*: Identifies the mailbox of the target application. No default. *MBX is mutually exclusive with XCP2WKS and is* ***mandatory*** *when XCP2WKS is omitted.*
X2 alias **XCP2WKS**	*X2=name8:* Identifies the local XCP2WKS which must be declared by an XCP2WKS directive. A local application using the current X2C can request to be associated to the XCP2WKS.
PARALLEL	Defines the type of session(s) between correspondents: *PARALLEL=1:* ***Default*** **-** Parallel sessions *PARALLEL=0:* only *single* sessions are supported.
PRIM alias **PRIMARY**	**Meaningful only if PARALLEL=0.** Determines the type of authorization for the single session: *PRIM=1:* ***Default*** **-** PRIMARY *PRIM=0: secondary*.

WIN alias **WINNER**

Meaningful only if PARALLEL=0. Determines how contention for the single session is resolved:

WIN=1: ***Default*** - *winner* session is allocated to a conversation and does not request acceptance from the correspondent.

WIN=0: loser session, the TP-Node must request acceptance from its correspondent to allocate the session to a conversation.

SYNCLVL alias **SYNC_LEVEL**

Specifies the synchronization services supported with the conversation by the correspondent.

SYNCLVL=CONFIRM: ***Default***
confirm and *confirmed* XCP2 services can be used:
- *confirm* service allows a program to request its partner to send a confirmation
- *confirmed* service allows a program to send a confirmation on receiving *confirm*.

SYNCLVL=SYNCPOINT: In addition to *confirm* and *confirmed* XCP2 services, the *syncpoint* and *backout* XCP2 services are available during the conversation.
- *syncpoint* service allows defining commitments for a transaction
- *backout* service allows rolling back the resources to their last successful *syncpoint* status.

USERCTL alias **USERID_CONTROL**

Meaningful only if PARALLEL=0. Defines the default option for security checks on the userid, when a transaction is initiated by the partner, and the single session has been established without opening the attached session pool.

USERCTL=OPTIONAL: ***Default*** **-** Userid, project and billing are optional:
- if specified, will be checked
- if omitted, the *already-verified* option taken.

USERCTL=MANDATORY: Userid, project and billing will ***always*** be checked. The *already-verified* option is ***ignored***.

USERCTL=NONE: The application does not accept security information. If such information is supplied, the request is rejected.

PROJCTL alias **PROJECT_CONTROL**

Meaningful only if PARALLEL=0. Defines the default option to be taken for security checks on access rights to the application by the project when a transaction is initiated by the partner, and the single session has been established without opening the attached session pool.

PROJCTL=1: ***Default***
Checks are made against the catalog.

PROJCTL=0: No checks

CORRESPONDENT PROFILE

Defined by parameters:

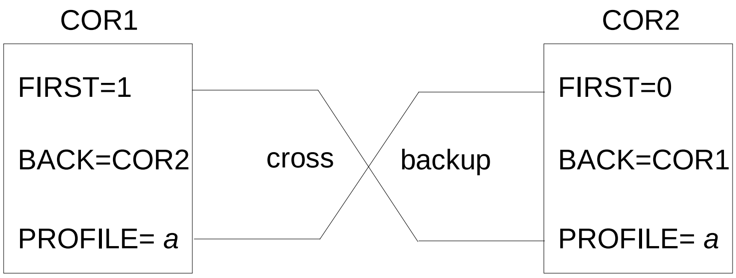
PARALLEL + PRIMARY + WINNER + SYNC_LEVEL + USERID_CONTROL + PROJECT_CONTROL

BACK alias **BACKUP**

BACK=name12: Identifies the backup correspondent and must be declared by another XCP2COR directive with:

– the same correspondent profile
– but the opposite FIRST attribute.

Example:
Two correspondents are declared, where COR1 is privileged and COR2 is the backup for COR1. The declaration for the respective correspondents is as follows:



FIRST alias **FIRSTBACK** alias **FIRST_BACKUP**

Meaningless when BACKUP is not specified.
Indicates if the correspondent in a pair of cross declared backups is privileged:
FIRST=1: ***Default*** **-** privileged
FIRST=0.

5.8 X2P (XCP2POOL) Directive

Purpose

This directive applies only to the directory configuration description and defines the attributes of a pool when opened by an XCP2 workstation. XCP2 sessions between two XCP2 workstations are described by one or more pools. The XCP2 pool groups parallel sessions having the same attributes. A pool cannot be defined for a single session.

A pool is identified by a name or *mode*, and the correspondent partner XCP2 workstation to which the pool is attached. Several pools may be attached to the same correspondent.

Security Parameters

CONV_USERID, CONV_CHECK, CONV_VERIFIED_ACCEPTED apply only for a remote XCP2 application. These parameters define how security checks, if applied, are to be performed when starting up a remote transaction. If these checks are successful, the remote transaction is eligible for startup. See 5.9 Conversation Security Mechanism.

Syntax

```
{ X2P      }                          [ { X2C     }         ]
{          }          NAME=name8      [,{         }=name12  ]
{ XCP2POOL }                          [ { XCP2COR }         ]

[ { X2C      }        ]          [ { MAXSC   } { 2    } ]
[,{          }=name8  ]          [,{         }={      } ]
[ { XCP2WKS  }        ]          [ { MAXSESS } { dec4 } ]

[ { WINSRCE           } { 1    } ]   [ { WINAUTO     } { 0    } ]
[,{                   }={      } ]   [,{             }={      } ]
[ { MIN_WINNER_SOURCE } { dec4 } ]   [ { WINNER_AUTO } { dec4 } ]

[ { WINTRGT           }   { 1    } ]
[,{                   } = {      } ]
[ { MIN_WINNER_TARGET }   { dec4 } ]

[ { DRSRCE       } { 0 } ]           [ { DRTRGT       } { 0 } ]
[,{              }={   } ]           [,{              }={   } ]
[ { DRAIN_SOURCE } { 1 } ]           [ { DRAIN_TARGET } { 1 } ]

[ { SYNCLVL    } { CONFIRM   } ]     [ { CONVCK     } { 1 } ]
[,{            }={           } ]     [,{            }={   } ]
[ { SYNC_LEVEL } { SYNCPOINT } ]     [ { CONV_CHECK } { 0 } ]
```

```
[ { CONV_X2SECU           }   { 1 } ]
[,{                       } = {   } ]
[ { CONV_XCP2WKS_SECURITY }   { 0 } ]

[ { CONV_ACCEPT             }   { 0 } ]
[,{                         } = {   } ]
[ { CONV_VERIFIED_ACCEPTED }   { 1 } ]

[ { CONV_USER   }   { OPTIONAL  } ]
[,{             } = { MANDATORY } ] ;
[ { CONV_USERID }   { NONE      } ]
```

Parameter Description

NAME *NAME=name8:* Identifies the XCP2 pool.

X2C alias **XCP2COR** *X2C=name12:* Identifies the *partner* XCP2 workstation:

- if omitted, the ***default*** is the name specified for X2C in an X2P directive previously declared
- if specified, must be defined in an X2C directive.

X2 alias **XCP2WKS** *X2=name8:* Identifies the XCP2 workstation to which the XCP2 pool is attached:

- if omitted, the ***default*** is NAME of an XCP2WKS directive previously declared
- if specified, must be defined in an XCP2WKS directive.

MAXSC alias **MAXSESS** *MAXSC=dec4:* Defines the maximum number of parallel sessions requested for the pool by the application. **The partner correspondent specified by X2C must declare the attribute PARALLEL=1 in its directive.**

If maxsess > 1, winauto <= maxsess >= winsrce + wintrgt.
0 < maxsess <= 5000. ***Default****: 2.*

WINSRCE alias **MIN_WINNER_SOURCE**

WINSRCE=dec4: Defines the minimum number of winner sessions requested for the source XCP2 workstation issuing the *open-pool* primitive, by the application.
maxsess >= winsrce + wintrgt0 <= winsrce <= 5000.
Default: 1.

WINTRGT alias **MIN_WINNER_TARGET**

WINTRGT=dec4: Defines the minimum number of winner sessions requested for the target XCP2 workstation receiving the *open-pool* primitive, by local application.

maxsess >= winsrce + wintrgt

0 <= *wintrgt* <= 5000. ***Default:*** 1.

WINAUTO alias **WINNER_AUTO**

WINAUTO=dec4: Defines the number of winner sessions activated by the local XCP2 workstation.

winauto <= maxsess >= winsrce + wintrgt

0 <= *winauto* <= 5000. ***Default:*** 0.

DRSRCE alias **DRAIN_SOURCE**

Determines how allocation requests issued by the ***local*** XCP2 workstation for that pool, are processed:

DRSRCE=0: ***Default***, allocation requests will be rejected when the pool is closed

DRSRCE=1: Allocation requests are accepted.

DRTRGT alias **DRAIN_TARGET**

Determines how allocation requests issued by the remote XCP2 workstation for that pool, are processed:

DRTRGT=0: ***Default***, allocation requests will be rejected when the pool is closed

DRTRGT=1: Allocation requests are accepted.

SYNCLVL alias **SYNC_LEVEL**

Specifies the synchronization services supported on the conversation by the correspondent:

SYNCLVL=CONFIRM: ***Default****, confirm and confirmed* XCP2 services can be used:
- *confirm* service allows a program to request its partner to send a confirmation
- *confirmed* service allows a program to send a confirmation after receiving *confirm*.

SYNCLVL=SYNCPOINT: In addition to *confirm* and *confirmed, syncpoint and backout* services are available during the conversation, in which case, ***the corresponding XCP2 workstation declared by XCP2WKS directive must declare SYNCPOINT=1***:
- *syncpoint* service allows defining commitments for a transaction
- *backout* service allows rolling back the resources to their last successful *syncpoint* status.

CONV_X2SECU alias **CONV_XCP2WKS_SECURITY**

Conversation security parameter, see paragraph 5.9.
Determines where the conversation security options, CONV_USER, CONV_ACCEPT and CONVCK, for this pool are to be taken:

CONV_X2SECU=1: ***Default***
From the corresponding XCP2WKS directive.

CONV_X2SECU=0: From the parameters in the directive itself.

CONV_USER alias **CONV_USERID**

Conversation security parameter, see paragraph 5.9.
Defines the remote security procedure:

CONV_USER=MANDATORY: The XCP2 application must perform conversation security checks depending on the setting of the *already-verified* flag in the incoming request:
- if OFF: PASSWORD must be provided and the request is then checked according to the CONV_CHECK parameter value; otherwise the request is rejected.
- if ON: the request is either rejected if CONV_VERIFIED_ACCEPTED=0 or checked according to the CONV_CHECK parameter value.

CONV_USER=NONE: No checks; the conversation is always accepted.

CONV_USER=OPTIONAL: ***Default****,* result depends on the contents of the incoming request:
- if USER value is specified, acts as CONV_USER=MANDATORY
- if USER value is omitted, acts as CONV_USER=NONE.

CONV_ACCEPT alias **CONV_VERIFIED_ACCEPTED**

Conversation security parameter, see paragraph 5.9.
Not applicable with CONV_USER=NONE.
Specifies whether an incoming request with an *already-verified* flag set to ON is to be accepted.

CONV_ACCEPT=0: ***Default*** **-** Rejected

CONV_ACCEPT=1: Accepted depending on the checks based on values of the CONV_CHECK parameter.

CONVCK alias **CONV_CHECK**

Conversation security parameter, see 5.9. Not applicable with CONV_USER=NONE.
Checks that the transaction can be invoked on behalf of the PROJECT.

CONVCK=1: ***Default***
The PROJECT must have in its site CATALOG this XCP2 application defined by MAILBOX of the associated XCP2WKS directive, in its application list and has the proper authority code to access it.

CONVCK=0: No project check.

5.9 Conversation Security Mechanism

Conversation Security affects a remote XCP2 application which is declared in the following directives:

- X2 (XCP2WKS)
- and X2P (XCP2POOL).

The parameters involved in both directives are:

- CONVCHK (CONV_CHECK)
- CONV_ACCEPT (CONV_VERIFIED_ACCEPTED)
- and CONV_USERID.

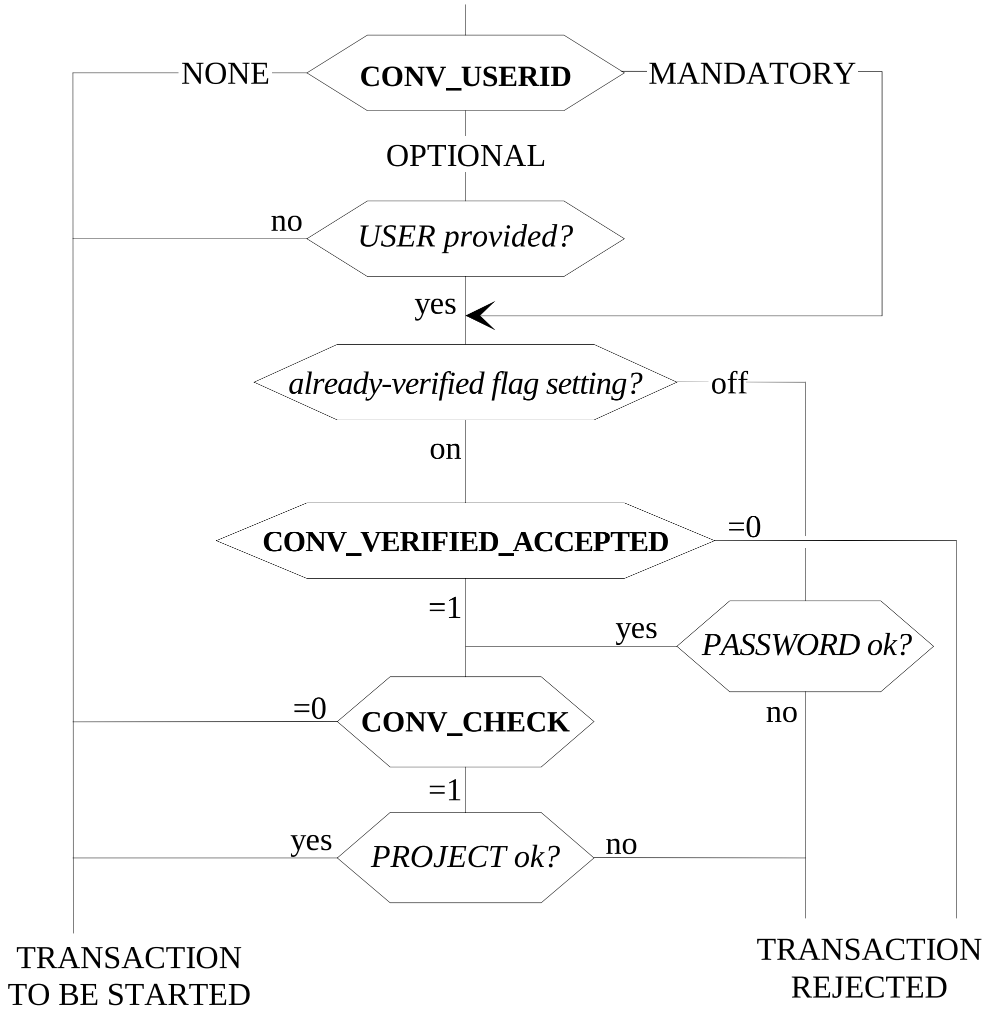
These parameters define *before starting a remote transaction*, if **security checks** have to be performed and how.

If these checks are successful the remote transaction becomes eligible to be started.

Checks are made on the following criteria:

- the *USER*, *PROJECT*, *BILLING* and *PASSWORD* cataloged in the SITE.CATALOG, with actual values provided by the user as required in the incoming invocation request
- and an internal *already-verified* flag which is set to *ON* when these incoming values have been checked by the invoking transaction (SAME mode).

The following chart summarizes the use of the three conversation security parameters within the conversation security checking process.

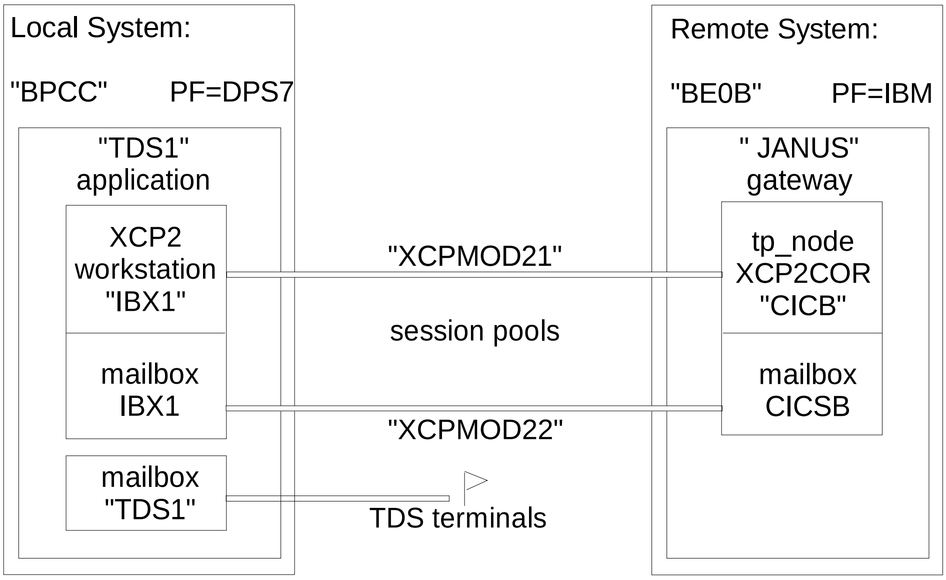

CONV_USERID
NONE
MANDATORY
OPTIONAL
no
USER provided?
yes
already-verified flag setting?
off
on
CONV_VERIFIED_ACCEPTED
=0
=1
yes
PASSWORD ok?
no
=0
CONV_CHECK
=1
yes
PROJECT ok?
no
TRANSACTION
TO BE STARTED
TRANSACTION
REJECTED

5.10 Examples of XCP2 Correspondents

EXAMPLE 1

I-SCHEMATIC DESCRIPTION

The example shows a local TDS application dialoging with a remote CICS application through XCP2 protocol and using two session pools:



❑

II-SERVICE OF CORRESPONDENTS DIRECTIVES

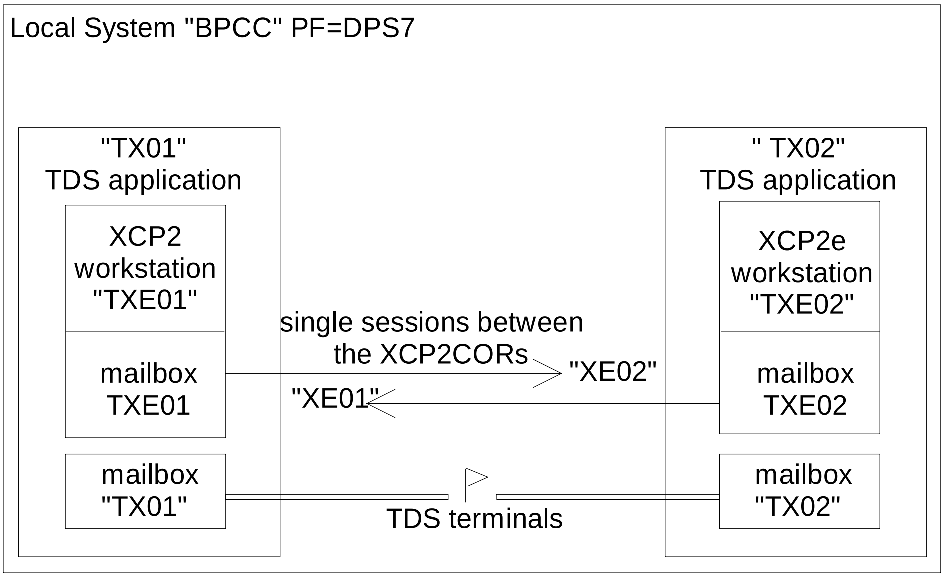
The associated directives, describing in the basic network configuration the two referenced systems, are not shown here.

```
DIR;

COMM '+---------------------------------------------+';
COMM '| COOPERATIVE PROCESSING BETWEEN THE TDS       |';
COMM '| APPLICATION TDS1 AND THE IBM CICS           |';
COMM '| APPLICATION CICB (REMOTE XCP2 CORRESPONDENT  |';
COMM '| CICB) THROUGH OSF GATEWAY (BE0B.CICSB) USING |';
COMM '| TWO SESSIONS POOLS (XCPMOD21, XCPMOD22)      |';
COMM '+---------------------------------------------+';

TDSWKS      NAME=TDS1
            TMSESS=5
            XCP2WKS=IBX1;

XCP2WKS     NAME=IBX1 ;

XCP2COR     NAME=TDSLOC1,
            XCP2WKS=IBX1;

XCP2COR     NAME=CICB,
            SESS=BE0B,
            MAILBOX=CICSB ;

XCP2POOL    NAME=XCPMOD21,
            XCP2COR=CICB,
            XCP2WKS=IBX1,
            MAXSESS=10,
            WINNER_AUTO=1 ;

XCP2POOL    NAME=XCPMOD22 ;

EDIR;
```

EXAMPLE 2

I-SCHEMATIC DESCRIPTION

The example shows two local TDS applications dialoging through XCP2 protocol and using the single session mechanism:



❑

II-SERVICE OF CORRESPONDENTS DIRECTIVES

The associated directives, describing in the basic network configuration the two referenced systems, are not shown here.

```
DIRECTORY;

COMM '+--------------------------------------------+';
COMM '| COOPERATIVE PROCESSING BETWEEN TWO TDS      |';
COMM '| APPLICATIONS (TX01 AND TX02)               |';
COMM '| THE TWO XCP2 CORRESPONDENTS (XE01 AND XE02),|';
COMM '| LOCAL TO THE DPS7,COMMUNICATE THROUGH A     |';
COMM '| "SINGLE" LOCAL VCAM SESSION.               |';
COMM '+--------------------------------------------+';

TDSWKS      NAME= TX01
            TMSESS=5
            XCP2WKS = TXE01;

TDSWKS      NAME= TX02
            TMSESS=5
            XCP2WKS = TXE02;

XCP2WKS     NAME=TXE01;

XCP2WKS     NAME=TXE02;

XCP2COR     NAME=XE01,
            XCP2WKS=TXE01,
            PARALLEL=0,
            PRIMARY=1;

XCP2COR     NAME=XE02,
            XCP2WKS=TXE02,
            PARALLEL=0,
            PRIMARY=0;

EDIR;
```

6. Low-Level Network Description

6.1 Low-level Description of the Primary Network

The usual network configuration description outlined in Section 3 uses only one directive per system. The low-level network configuration description hereafter explained uses one directive per layered object.

The information treated here assumes that the network configuration description uses only low-level directives.

Mixing SYSTEM directives and low-level ones is not recommended outside the scope of situations treated in Section 7.

For MCS applications, all remote systems which dialog with QMON must be declared in the Basic Network Configuration.

6.1.1 Local Objects

NOTE:
Where the DPS 7000 works in the OPEN 7 environment and communicates only through the TCP/IP protocols, only local objects need to be described.

The following objects relating to the Local System must be explicitly declared:

- LSYS:
  Provides the global attributes of the Local System in its profile defining its GCOS 7 hardware version and software release.
- LSC:
  The Session Control entity of the Local System which handles the DSA200 session protocol. An LSC is addressed by its TSAP whose format conforms to the ISO/DSA standard.

- SVR:
  Among the different types of communications servers, only FEPS occurrences, TNS and OCS must be explicitly defined with their specific attributes. Such attributes include the list of PSI channels connecting the corresponding Datanets. The PSI is declared by the LCT directive.

When the system provides an access to the ISL through TNS, the following additional objects must be defined:

- NR:
  Defines the logical path used by the Local Transport Station to access a Remote Transport Station. The NR can therefore be considered a local as well as a remote object. The endpoints of an NR are located on the same ISL cable which constitutes the Communications Path. The exit from the Local Transport Station is the local endpoint of the NR.
- CP:
  Provides the link between the local DPS 7000 and the ISL cable through the following objects:
  - LPL:
    the Local Physical Link of an LT to the ISL cable
  - LCT:
    a Local Controller which is a part of the SPA
  - LT:
    pairs of Logical Terminators (or channels) that link the LCT to the DPS 7000 through a PSI channel. When no LT is specified, default LTs are generated by the NETGEN utility.

When the system provides an access to FDDI or Ethernet through OCS Front End, the following additional object must be defined:

- LCT:
  Defines a local OCS Front End controller.

6.1.2 Remote Objects

The following objects relating to a system other than the local system must be explicitly declared:

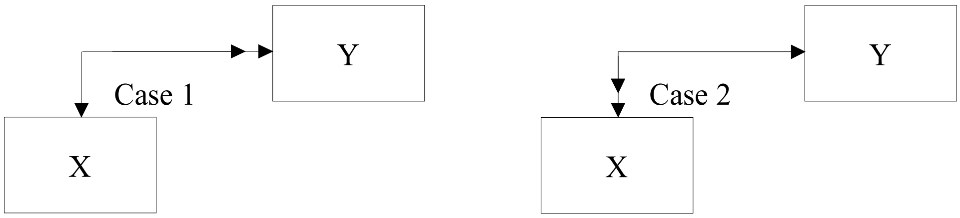
- RSYS:
  Represents the system's global attributes in its relation to the DPS 7000 (neighbor, passthrough or remote system) and profile defining its hardware version and software release. If the passthrough is administered by the local system, the name of the administering server is also specified.
- RSC:
  Each system has a remote session control entity associated with a server being either an FEPS occurrence for a Datanet, TNS for a CNP 7, or OCS for its front end.
- SR:
  The server establishes a session route towards the RSC. SRs can be explicitly defined and given attributes after the corresponding RSC is declared.
- Transport Connection:
  TNS establishes transport connections either directly with other DPS 7000 systems attached to the ISL or indirectly with any configurable system through the CNP 7 over the same ISL or another LAN, through the following objects:
  - RTS:
    For direct connections, the Transport Station of the remote DPS 7000 being the endpoint of the connection must be defined in the network generation of the local DPS 7000.
    For indirect connections, the RTS is not visible to the local DPS 7000 and is therefore not declared in the network generation of the local DPS 7000. It is declared instead in the SYSGEN of the CNP 7 which functions exactly like the Datanet. RTS attributes include the name of a TP and a list of NRs.
  - TP:
    The Transport Protocol for which its credit and its size of Data Unit are defined.
  - NR:
    The Network Route describes the path to the LSYS through a list of RPLs that provide the exit from the RSYS.
  - RPL:
    A Remote Physical Link is a transceiver on the ISL and is associated with an RCT.
  - RCT:
    A Remote Controller drives the remote ISL transceiver.

6.1.3 Mapping Primary Network Objects

The following diagrams illustrate the mapping of local objects on remote objects depending on whether the server is TNS or FEPS.

The convention used in showing the relationship of these objects is as follows:

- the *one-to-one* relationship is indicated by a single arrow
- the *one-to-many* relationship is indicated by a double arrow



Case 1 : X is mapped onto several occurrences of Y, while Y is mapped on a single occurrence of X.

Case 2 : This is the inverse of Case 1.

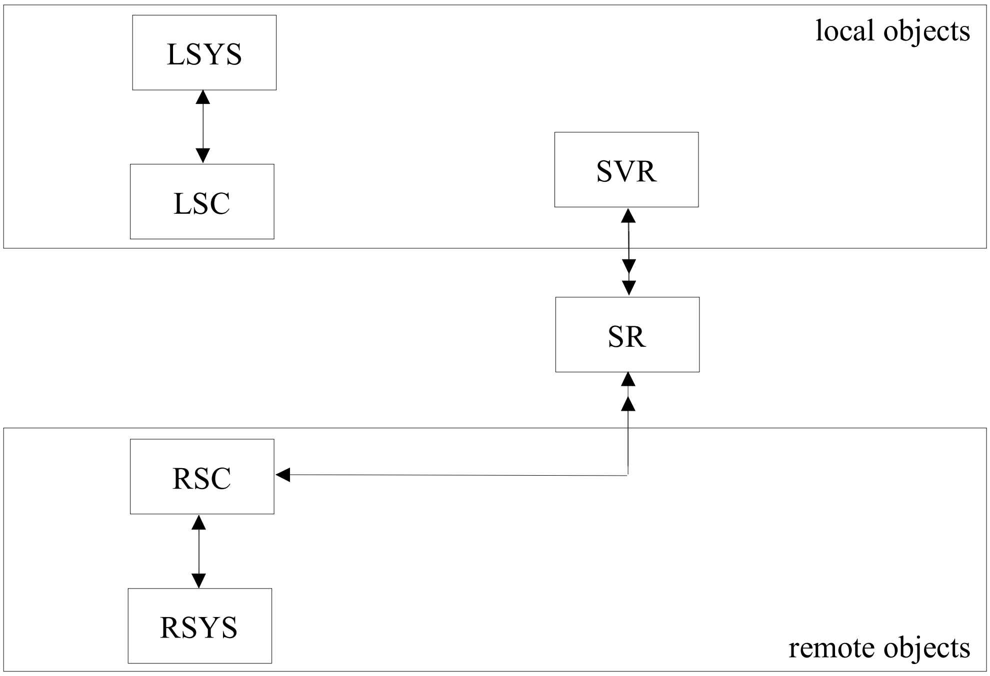


Figure 6-1. General Case of Object Mapping (Layers 5-7)

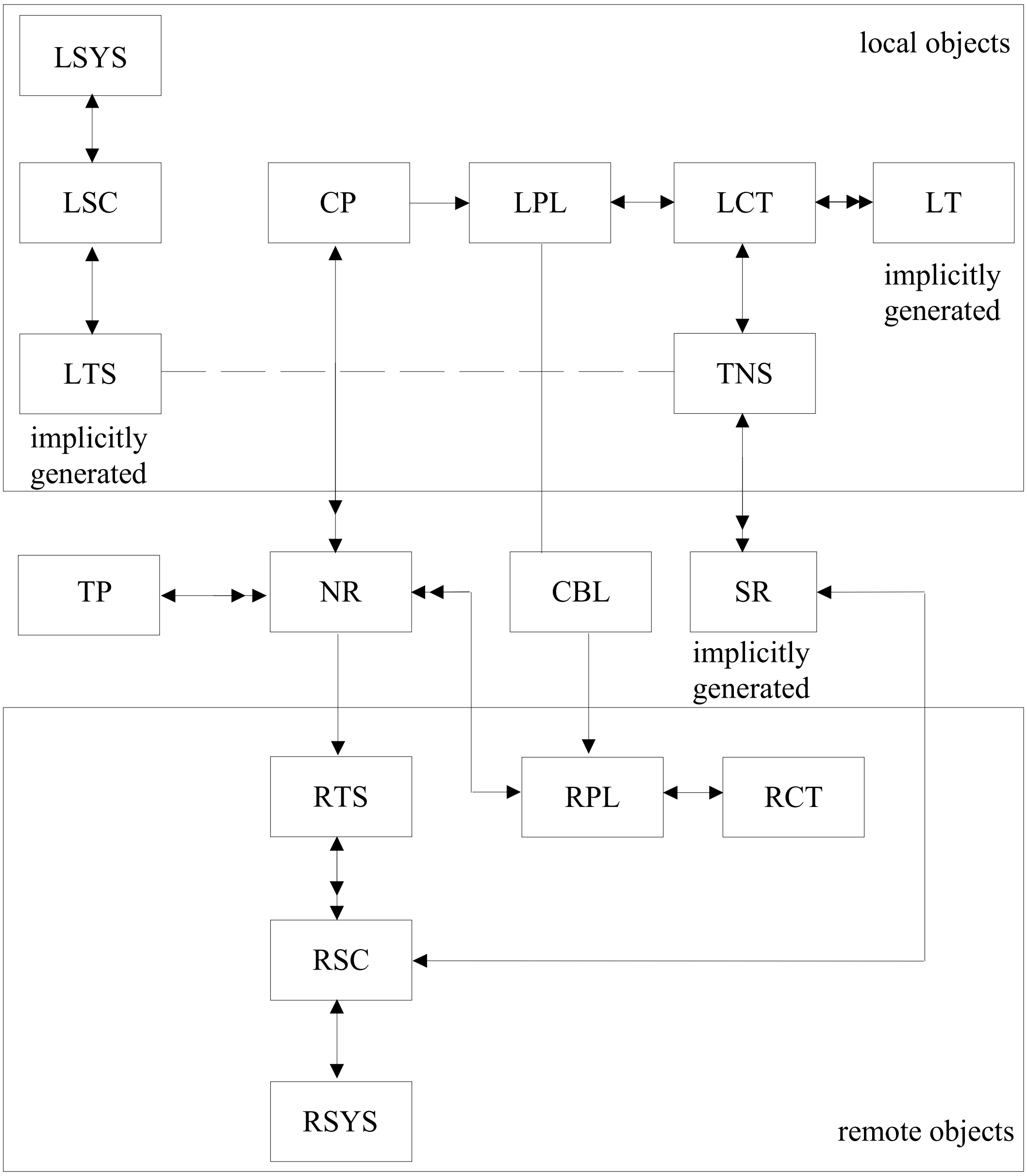


Figure 6-2. Mapping Objects for TNS

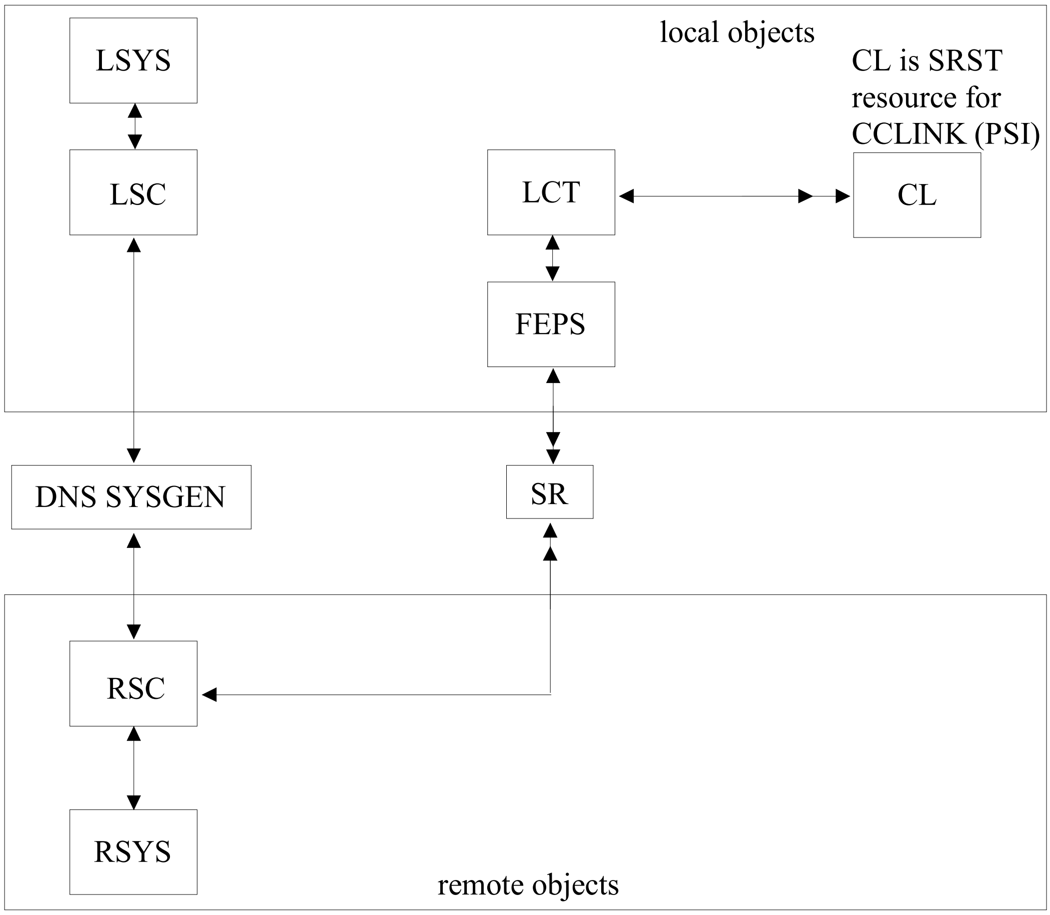


Figure 6-3. Mapping Objects for FEPS

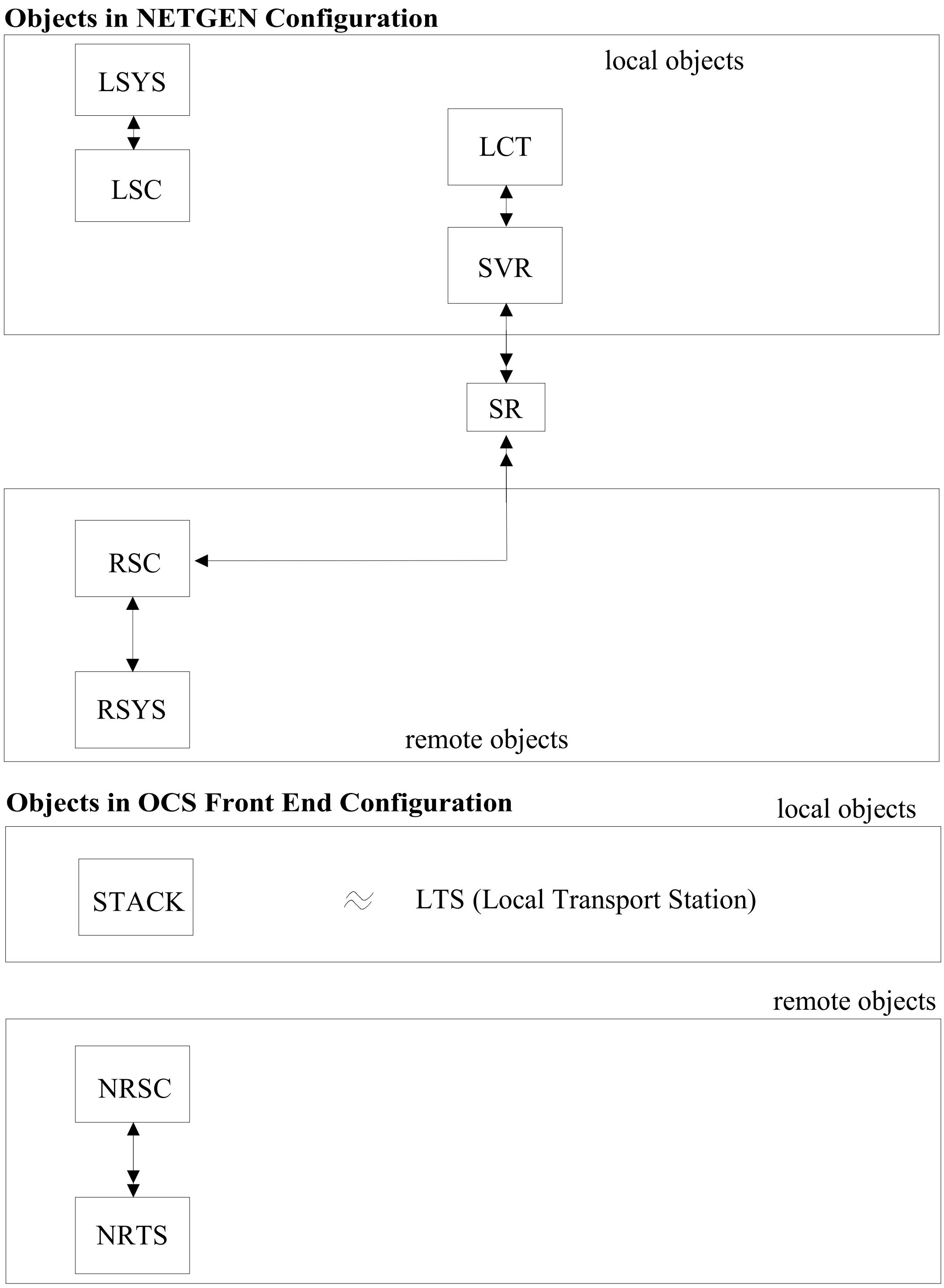


Figure 6-4. Mapping Objects for OCS

6.2 Low-level Directives

Syntax Rules for Low-level Directives

In the basic network configuration source, low-level directives may be entered in any order between the NET and ENET directives.

In the directory configuration source, low-level directives describing remote systems may be entered in any order between the DIR and EDIR directives. However, for ease of reference, they should be separated from directives declaring the service of correspondents which also belong to the directory configuration. See Section 5.

The sequence in which NETGEN directives are declared does not affect the access time for any object but it does affect the order in which the objects are listed.

The DNET command lists network objects:
- either in the same order as they were declared if these objects concern TNS
- or in the reverse order for directory objects, systems and their session controls.

OPEN 7 ENVIRONMENT

Where the DPS 7000 works in the OPEN 7 environment and communicates using only TCP/IP protocols, only the local system, described by low-level directives, need be declared in the basic network configuration description.

The directory description can therefore contain only correspondent objects related to the local system and local applications.

NOTES:

1. For all directives, parameters can be specified in any order. All arguments are introduced by their corresponding keyword. NAME is the only exception being the first positional and can therefore be omitted.
2. Several different objects may have the same name. Up to 8 identical names are allowed. In general, since each directive such as LSYS, LSC, RPL, LPL and SVR, corresponds to only one object, only one name is possible. The exception is the RSYS directive, which may generate up to 3 objects of the same name.
3. In any directive when the argument is boolean (0 or 1), the keyword on its own is equivalent to specifying 1 as its argument, for example:
   GENQMON is equivalent to GENQMON=1,
   NGENQMON is equivalent to GENQMON=0.

List of Low-level Directives

In this Section, the low-level directives are described in the alphabetical order of their abbreviations. The table below lists the directives in the order of their *unabbreviated form* with:

- the corresponding abbreviation in column ***Abbr***
- the configuration description destination in column ***Dest***: (B for Basic, D for Directory, Ds for remote systems)
- the reference of the directive
- the indication by an asterisk that the statement belongs to the set of low-level directives possibly generated by the SYSTEM directive in column ***HL***.

Mixing low-level directives with SYSTEM directives must strictly conform to the rules described in Section 7.

Abbr	Extended	Dest	Par.	HL	Abbr	Extended	Dest	Par.	HL
CP	COMPATH	B	6.3	*	RCT	RMTCTLR	B	6.11	*
LCT	LOCCTLR	B	6.4	*	RPL	RMTPLINK	B	6.12	*
LMBX	LOCMAILBOX	B	6.5		RSC	RMTSESS	BDs	6.13	*
LPL	LOCPLINK	B	6.6	*	RSYS	RMTSYSTEM	BDs	6.14	*
LSC	LOCSESS	B	6.7	*	RTS	RMTTRANSPORT	B	6.15	*
LSYS	LOCSYSTEM	B	6.8	*	SR	SESSROUTE	B	6.16	
LT	LTERMINATOR	B	6.9	*	SVR	SERVER	B	6.17	*
NR	NETROUTE	B	6.10	*	TP	TPROTOCOL	B	6.18	*

The following table gives the reference to the low-level directives for the basic network and directory configuration described in Section 3.

Abbr	Extended	Dest	Abbr	Extended	Dest	Abbr	Extended	Dest
COMM	COMMENT	BD	ENET	ENDNETWORK	B	NET	NETWORK	B
DIR	DIRECTORY	D	EDIR	ENDDIRECTORY	D	SDOM	SUBDOMAIN	Ds

6.3 CP (COMPATH) Directive

Purpose

This directive applies only to the basic network configuration description. The CP directive describes the physical path provided by one ISL transceiver.

Syntax

```
{ CP      }
{         }        NAME = name8
{ COMPATH }

 { LPL      }
,{          } = name8 ;
 { LOCPLINK }
```

Parameter Description

NAME

NAME=name8: Identifies the local name of the CP. No default.

LPL alias **LOCPLINK**

LPL=name8: Identifies the local physical link defined in a LPL directive. No default.

6.4 LCT (LOCCTLR) Directive

Purpose

This directive applies only to the basic network configuration description. It describes a local communications controller which is either a local ISL adaptor (LNA or LNM) to access the CNP 7 or other systems over the ISL, or PSI controller to access a Datanet over the PSI, or OCS Front End controller to access an FDDI or high performance Ethernet cable.

In the case of a local ISL adaptor (LNA or LNM), its name must be specified as the argument of the LCT parameter in the LPL directive.

Syntax Format 1 (ISL Controller)

```
{ LCT     }
{         }          NAME = EAhexa2         [, TYPE = ISL ]
{ LOCCTLR }

[ { LT          }                           ]
[,{             } = ( name8 [, name8 ]... ) ]
[ { LTERMINATOR }                           ]

[ { ST    }   { ENBL } ]
[,{       } = {      } ]
[ { STATE }   { LOCK } ]

[ { RESET    }   { 0 } ]
[,{          } = {   } ]
[ { RMTRESET }   { 1 } ]

[ { TCPIP     }                         ]
[,{           } = ( name6 [, name6 ] ...) ] ;
[ { TCPIPTYPE }                         ]
```

Syntax Format 2 (PSI Controller)

```
{ LCT     }
{         }           NAME = CChexa2          ,TYPE = PSI
{ LOCCTLR }

[ { CN    }    { 2000 } ]
[,{       } =  {      } ]
[ { MAXCN }    { dec5 } ]
[ { CH    }    { 16   } ]
[,{       } =  {      } ]
[ { MAXCH }    { dec2 } ]

[ { INCH    }    { 8    } ]
[,{         } =  {      } ]
[ { MAXINCH }    { dec2 } ]

[ { T3    }    { 320  } ]
[,{       } =  {      } ]
[ { WATCH }    { dec4 } ]

[            { 1 } ]
[,ADMLOAD =  {   } ] ;
[            { 0 } ]
```

Syntax Format 3 (OCS Front End Controller)

```
{ LCT     }
{         }  NAME = EAhexa2
{ LOCCTLR }

            ,TYPE = NCC

            ,IADDR = [{G | L}:] xx-xx-xx-xx-xx-xx

            [,TREQ = dec-3]

            [,MAXCN = dec-5]

            [,WATCH = dec-4]

            [,OP7_MAXCCE = (dec-3 [,dec-3])]

            [,OP7_TIMER = dec-3]
```

Parameter Description

NAME

Identifies the LCT which must be registered in the SRST table of the targeted local system.

NAME=EAhexa2: For ISL type, no default.

NAME=CChexa2: For PSI type, no default.

NAME=EAhexa2: For local OCS Front End controller, no default.

TYPE

Identifies the link connecting the LCT to the local DPS 7000.
TYPE=ISL: ***Default***
TYPE=PSI: Connection to the Datanet.
TYPE=NCC: Connection to OCS Front End.

IADDR

Only for TYPE = NCC, it is the individual address (provided at installation time) of the corresponding adapter or transceiver. The address has the following format:

`xx-xx-xx-xx-xx-xx`

where `xx` are hexadecimal.

This address may be prefixed by `L:` or `G:`
– `L:` means local (not unique in the network).
– `G:`, the default, means global (unique in the network).

The second part of the parameter identifies the local OCS Front End as declared in the SRST. The format is EAxx.

TREQ

For FDDI only, 4<=dec3<=168. Default is 10.
Unit=milliseconds.
It is the requested value for the Target Token Rotation time (TTRT) on the FDDI ring.

LT alias **LTERMINATOR** **Applicable only to ISL type.**

LT=(name8 [,name8]...)]: Lists up to 7 dedicated input and output logical channels:

up to 7 normal data traffic LTs (ISO/DSA or TCP/IP transport), and no remote maintenance LT

1 remote maintenance, and up to 6 normal data traffic LTs.

The LTs, each identified by up to 8 alphanumeric characters, handle communication over the PSI between:
TNS transport (ISO/DSA or TCP/IP)
and the local controller defined by the LCT directive.

Note:
All other LTs, administrative and foreign, are internally supplied by the system and cannot be explicitly generated. If TCP/IP transport is used, data and remote maintenance LTs must be explicitly declared.
If only ISO/DSA transport is used, defaults for data and remote maintenance LTs are generated by the system, namely, two data LTs and one RM LT. See LT directive for more details.

ST alias **STATE** **Applicable only to ISL type**.

ST=ENBL: ***Default*** **-** the LCT is activated at TNS startup.
ST=LOCK: The LCT requires operator intervention through the MDNET operator command to function.

RESET alias **RMTRESET** **Applicable only to ISL type**.

RESET=1: The LCT can be reset remotely.
RESET=0: ***Default***

MAXCN alias **CN** **For PSI type**, represents the maximum number of connections handled over the PSI.

1 <= *maxcn* <= 32767. **Default:** 2000.

For OCS Front End, represents the maximum number of transport connections handled by this OCS Front End.
1 <= MAXCN <= 5000. **Default:** 2000.

CH alias **MAXCH**

Applicable only to PSI type. Represents the maximum number of logical channels to be used over the PSI.

5 <= *maxch* <= 16. ***Default:*** 16.

ADMLOAD

Applicable only to PSI type.
ADMLOAD = 1 means that the front-end may be administered by the host (FECM). ADMLOAD = 0 means that the server (FEPS) can be launched but the front-end is not administered by the host.
Default: 1.

INCH alias **MAXINCH**

Applicable only to PSI type. Represents the highest index value for logical channels handling input data over the PSI.

3 <= *maxinch* <= (*maxch* - 2). ***Default:*** 8.

Example: MAXCH=16, MAXINCH=5 being the LC index

LC Indexes				
0	1	2	3 4 5	6 7…15
attention	command		data	
	input	output	input	output

T3 alias **WATCH**

For PSI type, represents the watchdog delay in tenths of a second between FEPS and the Datanet over the PSI. This value must match in the DNS configuration the T3 parameter value of the CH directive.
T3=0 means no watchdog timer.
200 <= *watch* <= 6000. ***Default:*** 320.

WATCH

For OCS Front End, defines the watchdog delay between OCS and the front end. The unit is 1/10 sec. The default is 320.
200 <= WATCH <= 6000.

TCPIP alias **TCPIPTYPE**

Applicable only to ISL type. Lists up to 7 transport protocols supported for which TCP/IP transport is used. When TCPIP is specified, the LT parameter becomes mandatory.

`TCPIP=IP:` Internet Protocol frames

`TCPIP=ARP:` Address Resolution Protocol frames

`TCPIP=RARP:` Reverse Address Resolution Protocol frames

`TCPIP=TRAILn:` Trailing TCP/IP header protocol frames. The size of data transmitted in the frame is *n* * 512 bytes, where 0 <= *n* <= 3.

Note:

Specifying LCT with TCPIPTYPE allows the local system to be the only system declared in the basic network configuration.

When TCPIP is omitted, ISO/DSA transport is assumed.

If TCPIP=1 is specified in the high-level SYSTEM directive, all the protocols are assumed to be supported by default.

OP7_MAXCCE

Applicable only to OCS Front End with OPEN 7. Represents the relationship between READ CCEs and WRITE CCEs (CCE = Channel Command Enters). OP7_MAXCCE = (dec-3,dec-3), where 1 <= dec-3 <= 100.
The default is OP7_MAXCCE = (1,6).
The first dec-3 value is the maximum number of READ CCEs concatenated. The second dec-3 value is the maximum number of WRITE CCEs concatenated. If only one value is specified, it is assumed to be the maximum number of both READ and WRITE CCEs.

OP7_TIMER

Applicable only to OCS Front End with OPEN 7. Represents the time limit for holding incoming frames for delivery to OPEN 7.
1 <= OP7_TIMER <= 100. Unit = milliseconds.
The default is OP7_TIMER = 10.
If all the linked buffers (defined by OP7_MAXCCE) are full (maximum number of READ CCEs received), or if the time specified by OP7_TIMER is reached (counted from the first incoming frame not yet delivered to OPEN 7), the OCS Front End notifies OPEN 7 (so that OPEN 7 reads the frames in the buffers).

6.5 LMBX (LOCMAILBOX) Directive

Purpose

This directive applies only to the basic network configuration description. It defines a DSA mailbox used to access a local application such as TDS. This directive applies when a terminal in the network is declared automatically assigned to the mailbox at DNS/CNS 7 generation. The mailbox allows session control to enqueue a connection request from the terminal whatever the state of the application. When the application is started up, it activates the mailbox and connects the terminal.

The maximum number of enqueued connections is 255.

Syntax

```
{ LMBX       }
{            }          NAME = name8 ;
{ LOCMAILBOX }
```

Parameter Description

NAME

NAME=name8: Identifies the mailbox which must be unique and cannot start with $.

It also cannot be one of the following system mailboxes:
IOF, RBF, READER, FILR, FILETRAN.

No default.

6.6 LPL (LOCPLINK) Directive

Purpose

This directive applies only to the basic network configuration description.It describes the local ISL transceiver.Values for MADDR, RADDR and FMADDR must only be specified on the advice of the installation personnel and on the recommendation of the network administrator.

Syntax

```
{ LPL      }
{          }        NAME = name8
{ LOCPLINK }

 { LCT     }
,{         } = EAhexa2
 { LOCCTLR }

         [ { G }   ]
,IADDR = [ {   } : ] xx-xx-xx-xx-xx-xx
         [ { L }   ]

 { CB    }
,{       } = name8
 { CABLE }

[ { ST    }   { ENBL } ]
[,{       } = {      } ]
[ { STATE }   { LOCK } ]

[            {G}                                                  ]
[, MADDR = {[{ } : ] xx-xx-xx-xx-xx-xx, xx-xx-xx-xx-xx-xx [,...]}]
[            {L}                                                  ]

[          {   00-00-00-00-00-00/7E-FF-FF-FF-FF-FF    } ]
[,RADDR =  {                                          } ]
[          { ( xd-xx-xx-xx-xx-xx/xd-xx-xx-xx-xx-xx )  } ]

[            [{ G } ]          [{ G } ]                       ]
[,FMADDR = ( [{   }:]ISL-addr/[{   }:]ISL-addr [, .../... ] ) ] ;
[            [{ L } ]          [{ L } ]                       ]
```

Parameter	Description
NAME	*NAME=name8:* Identifies the local ISL transceiver. No default.
LCT alias **LOCCTLR**	*LCT=EAhexa2:* Identifies the corresponding local ISL adaptor (LNA) included in the local SPA or LNM and described in a LCT directive.
IADDR	Identifies the Individual ADDRess, provided at installation time, of the corresponding transceiver on the ISL cable. ***DPS 7000****:* *IADDR=[{L|G}:]08-00-38-dx-xx-xx*, ***DPS 7000/XTA****: IADDR=[{L|G}:] xx-xx-xx-xx-xx-xx,* where: *L:* means local and *G:* means global. ***Default****:* G - the address is network-wide and unique. 08-00-38 is the upper part reserved to BULL S.A. 1 <= *d* <= 7 and *xx* is hexadecimal.
CB alias **CABLE**	*CB=name8:* Identifies the external name of the ISL cable to which the transceiver is attached. No Default.
ST alias **STATE**	Indicates when the link will be active: *ST=ENBL:* ***Default*** As soon as the related LCT state is ENBL. *ST=LOCK:* The link will function only by operator intervention through the MDNET operator command.
MADDR	Lists up to 8 Multicast ADDResses. *MADDR=([{L|G}:]xx-xx-xx-xx-xx-xx,char19)*, where *L:* means local and *G:* means global. *xx* are hexadecimal digits *char19 represents {L|G}:xx-xx-xx-xx-xx-xx*. ***DPS 7000 : Default****:* G:08-00-38-10-00-00,G:08-00-38-20-00-00, where G: the address is network-wide and unique 08-00-38 is reserved for BULL S.A. (DPS 7000 system only)

RADDR

Lists up to 12 pair-sets of remote individual addresses. A pair-set represents a range of addresses, the first being the lower address, the second the upper. The addresses are separated by a /.
RADDR=(xd-xx-xx-xx-xx-xx,xd-xx-xx-xx-xx-xx), where
xx are hexadecimal digits
d takes even values: 0, 2, 4, 6, 8, A, C or E.
08-00-38 is reserved for BULL S.A (DPS 7000 system only).

***Default**:* 00-00-00-00-00-00/7E-FF-FF-FF-FF-FF

FMADDR

Lists up to 10 pair-sets of Foreign Multicast ADDResses, each pair-set being a range of addresses, the first the lower, the second the upper. The addresses are separated by a /.
FMADDR=([{L|G}:]ab-cd-ef-uv-wx-yz,char39...), where
L: means local and *G:* means global.

***Default**:* G: - the address is network-wide and unique.

ab-cd-ef must be different from *08-00-38*
u,v,w,x,y,z are hexadecimal digits.

6.7 LSC (LOCSESS) Directive

Purpose

This directive applies only to the basic network configuration description. This directive is mandatory and must be unique. It defines:

- the DSA address of the local system through the SCID parameter.
- the OSI address of the local system through the TSEL and NSAP parameters.
- the local PID module through the PIDTSEL and NSAP parameters. When omitted a default PIDTSEL is assumed.

OSI Syntax Format

```
{ LSC     }
{         }          [,NAME = name4 ]
{ LOCSESS }

 { ADDR     }
,{          } = ISO  ;
 { ADDRTYPE }

[ { PRTC     }                ]
[,{          } = { SID | ISO } ]
[ { PROTOCOL }                ]

[ { TSEL      }          ]
[,{           } = hexa64 ]
[ { TSELECTOR }          ]

[ { PID     }   { PID_SELECT } ]
[,{         } = {            } ]
[ { PIDTSEL }   { hexa64     } ]

[,NSAP = ( hexa40 [ hexa40 ]... ) ];

,SCID = dec-3 : dec-3 ;
```

DSA Syntax Format

```
{ LSC     }
{         }         [,NAME = name4 ]
{ LOCSESS }

[ { ADDR     }       ]
[,{          } = DSA ]
[ { ADDRTYPE }       ]

[ { PRTC     }   { DSA | DSA200 } ]
[,{          } = {              } ]
[ { PROTOCOL }   { SID | ISO    } ]

,SCID = dec3 : dec3 ;
```

Parameter Description

NAME

NAME=name4: Identifies the Session Control of the Local System. When omitted the default LSC name is:

either the name in the LSYS directive appearing immediately before the LSC directive
or the name in the NETWORK directive.

The names for the LSYS and LSC directives must be identical.

ADDR alias **ADDRTYPE**

Specifies if ISO or DSA format is to be used for session addressing and syntax. When omitted the default addressing type is:

*ADDR=ISO: **Default**, if TSEL or NSAP or PIDTSEL is specified.*

*ADDR=DSA: **Default**, if TSEL, NSAP and PIDTSEL are absent.*

Note:
In the current Release even with ISO, the local system needs to be addressed in both OSI and DSA to be accessed.
The values of ADDRTYPE in the LSC and LSYS directives are not cross checked. Even so, it is recommended for both values of ADDRTYPE to be identical.

PRTC alias **PROTOCOL** Specifies if the transmission protocol is ISO/DSA or DSA:
PRTC= ISO or SID: **Default**, if ADDRTYPE=ISO.
PRTC= DSA or DSA200: ***Default****, if ADDRTYPE=DSA.*

TSEL alias **TSELECTOR** *TSEL=hexa64:* **Applicable only to ISO Syntax Format.** No default. When the number of digits is odd, TSEL is internally right padded by one zero. With TSEL, local ISO applications provide only the SSEL when calling the SSAP address, namely:

- the session layer services use the configured TSEL as the default calling TSEL
- and the transport layer services in the Datanet use its configured object to set the default calling NSAP.

PID alias **PIDTSEL** **Applicable only to ISO Syntax Format.** Represents the TSELECTOR part in the OSI addressing of the local PID module.

PID=hexa64: Specifies the transport layer selector for the OSI addressing of the local system PID.

PID=PID_SELECT: ***Default***. Specifies a default transport layer selector for the OSI addressing of the local system PID. Generates the value "5049445F53454C454354" in ASCII.

NSAP **Applicable only to ISO Syntax Format.**
Allows the DPS 7000 to know that the address of a called ISO application is local, without having to access the Datanet. This occurs when the calling and called applications are both local and the one called has declared itself to the session layer without specifying an NSAP.

NSAP=(hexa40,hexa40...): Lists up to 20 Network Service Access Points attached to the local system and used to identify local connections. No default.

SCID Specifies the DSA address of the local session control.

SCID=dec3:dec3, where 0 <= *dec3* <= 255.

No default. Since the administration of the DPS 7000 and its FEP continues to be based on DSA objects and addressing, SCID is mandatory even with OSI syntax format.

6.8 LSYS (LOCSYSTEM) Directive

Purpose

This directive applies only to the basic network configuration description. This command is mandatory and must be unique.

It describes the local system for which the network configuration is generated.

ISO Syntax Format

```
{ LSYS      }                               [ { ADDR     }     ]
{           }       [,NAME=name4 ]          [,{          }=ISO ]
{ LOCSYSTEM }                               [ { ADDRTYPE }     ]

[ { PF      }   {                                 } ]
[,{         } = { DPS7 | 'DPS7/GCOS-7/{V9|V10}'   } ]
[ { PROFILE }   {                                 } ]
```

DSA Syntax Format

```
{ LSYS      }                               [ { ADDR     }     ]
{           }       [,NAME=name4 ]          [,{          }=DSA ]
{ LOCSYSTEM }                               [ { ADDRTYPE }     ]

[ { PF      }   {                                 } ]
[,{         } = { DPS7 | 'DPS7/GCOS-7/{V9|V10}'   } ]
[ { PROFILE }   {                                 } ]
```

Parameter Description	
NAME	*NAME=name4:* Identifies the local system as network-wide and unique. When omitted the default LSYS name is: either the name in the LSC directive appearing immediately after the LSYS directive or the name in the NET directive. **Note:** The LSYS and LSC directives must have identical names. The name in the NETGEN LSYS directive must be identical to the name declared in the CONFIG LSYS statement of the DPS 7000 for which the generation is performed and later enabled. Both these names are compared and enabling will be rejected if the two names are not identical.
ADDR alias **ADDRTYPE**	Specifies if the local system is accessible by either both ISO and DSA addressing, or DSA addressing only. *ADDR=DSA:* ***Default*** *ADDR=ISO* **Note:** In the current Release even with ISO, the local system needs to be addressed in both OSI and DSA to be accessed. The values of ADDRTYPE in the LSC and LSYS directives are not cross checked. Even so, it is recommended for both values of ADDRTYPE to be identical.
PF alias **PROFILE**	Defines the local system's hardware and software characteristics to allow UFT and DJP applications to identify its capabilities. See DPS 7000 file transfer capabilities in the RSYS directive. The profile can be specified in either the abbreviated or extended form. *PF=DPS7 or DPS7/GCOS-7/V10:* ***Default***

Network Security Control Service

The local system can be declared *secured*, in which case security checks are applied on all incoming connections from remote systems and on all connections between local applications.

For security checks to be performed on incoming connections:

- the local system must be declared *secured* in the network configurations of each passthrough front-ending the local system and dedicated to its security
- the security server must be known to and accessible by all the passthroughs front-ending the local system and dedicated to its security.

For security checks to be performed on local connections:

- the local system must be locally declared *secured* by specifying NETSEC=YES in the SECOPT directive of the GCOS 7 configuration
- all local connections must be *internally transformed by the local VCAM session server into pseudo-incoming connections. This means that the connection is considered to be from the local system towards itself, via a pseudo-remote system with the reserved name BCLS*
- The pseudo-remote system BCLS must be declared in the basic network configuration as follows:
  - in the RSYS: name must be BCLS, and the values for PROFILE and SCID must be those of the local system
  - in the RSC: name must be BCLS, and the list given in SVR and RTS must map onto only the passthroughs dedicated exclusively to the access security of the local system, and no other passthrough.

At enabling time, the security option selected at GCOS 7 CONFIG is checked for consistency against the NETGEN declaration as follows:

- if security is set at CONFIG but no BCLS has been declared in the NETGEN, the inconsistency is flagged by *severity 4 error NG086*
- if security is not set at CONFIG, local connections will not undergo security checks, in which case the pseudo-system BCLS is meaningless.

6.9 LT (LTERMINATOR) Directive

Purpose

This directive applies only to the basic network configuration description. It defines the input-and-output pairs of logical channels used for the communication between TNS and one local controller defined by a LCT directive with TYPE=ISL. This directive is optional when only ISO transport is used but is mandatory with TCP/IP transport.

There can be up to eight LTs which belong to 5 types depending on the kind of traffic they are dedicated to handle:

- *ALT:* Dedicated to the administration with the LCT. *It is internally generated and therefore cannot be explicitly declared.*
- *FLT:* Dedicated to foreign frames. *If required, it is installed by the BULL service center and is not normally customer visible.*
- *RMLT:* Dedicated to the administration and remote maintenance of the CNP 7. RMLT is mandatory if at least one CNP 7 is serviced by the local TNS and FECM servers:
  - either the RSYS directive is specified with RCT different from NO
  - or the SYSTEM directive is specified with LOAD=1.

  An RMLT is internally generated if:
  - either the related LCT directive does not explicitly refer to an LT and TCP/IP transport is not used
  - or the local system is declared by a SYSTEM directive.

  Otherwise, the RMLT must be explicitly declared by an LT directive with LSAP=RM.
- *DATA LT:* Dedicated to normal data and generated as follows:
  - *If TCP/IP transport is used:*
    1 TCP/IP data LT must be explicitly declared with LSAP=TCPIP.
  - *If TCP/IP and ISO/DSA transport are used:*
    One or more ISO/DSA data LTs must be declared through LT directives with LSAP=ISO. At least one of each type, both types totalling a maximum of 6.
  - *If the local system is declared by a high-level SYSTEM directive with TCPIP:*
    One data LT with LSAP=TCPIP is internally generated.
  - *If only ISO/DSA transport is used:*
    Two ISO/DSA data LTs are internally generated for a given local controller if:

either its LCT directive does not refer to any LT
or the local system is declared by a SYSTEM directive.

Otherwise up to 7 data LTs must be explicitly declared by LT directives with LSAP=ISO.

Syntax

```
{ LT          }                          [ { TIMER      }   { 10    } ]
{             }     NAME=name8           [,{            } = {       } ]
{ LTERMINATOR }                          [ { READTIMER }    { dec-3 } ]

  {      { RM  } [ { MAXCCE  }     { 3    } [  { 3    }    ] }
  { LSAP={     } [,{         } = ( {      } [,  {      } ) ] }
  {      { ISO } [ { MAXREAD }     { dec3 } [  { dec3 }    ] }
, {                                                          }
  {              [ { MAXCCE  }     { 1    } [  { 6    }    ] }
  { LSAP = TCPIP [,{         } = ( {      } [,  {      } ) ] }
  {              [ { MAXREAD }     { dec3 } [  { dec3 }    ] }
```

Parameter Description

NAME *NAME=name8:* Identifies the Logical Terminator.

LSAP Defines the type of LT:

`LSAP=RM:` LT dedicated to Remote Maintenance.
`LSAP=ISO:` LT dedicated to ISO data.
`LSAP=TCPIP:` LT dedicated to TCP/IP data.

MAXCCE alias **MAXREAD**

Represents the relationship of READ to WRITE CCEs.

MAXCCE=(dec3,dec3), where 1 <= *dec3* <= 100 and:

- the first value being the maximum number of READ CCEs concatenated
- the second value being the maximum number of WRITE CCEs concatenated
- if only one value is specified, READ CCEs and WRITE CCEs are assumed to be equal in number.

Default*:* MAXCCE=(3,3) for LSAP = ISO or RM.
Default*:* MAXCCE=(1,6) for LSAP = TCPIP.

TIMER alias **READTIMER** Specifies the timeout in milliseconds from the first incoming frame not yet delivered to TNS, after which the controller issues a notification to TNS.

Note:
The controller also issues a notification to TNS, if all the linked buffers defined by MAXREAD are filled by incoming frames.
5 <= *timer* <= 100, ***Default**:* 10 milliseconds.

RESPONSE TIME TUNING

Increasing TIMER and MAXREAD values reduces CPU consumption but produces a longer response time.

Reducing the TIMER value optimizes the response time, but decreases the blocking ratio whatever the number of CCEs specified by MAXREAD.

6.10 NR (NETROUTE) Directive

Purpose

This directive applies only to the basic network configuration description. It defines the logical path used for data transiting via an ISL between the local and remote transport stations.

Syntax

```
{ NR       }
{          }           NAME = name8
{ NETROUTE }

 { CP      }
,{         } = name8
 { COMPATH }

 { RPL      }
,{          } = name8
 { RMTPLINK }

 { TP        }
,{           } = name8 ;
 { TPROTOCOL }
```

Parameter Description

NAME	*NAME=name8:* Identifies the network route known only to the local system.
CP alias **COMPATH**	*CP=name8:* Identifies the communications path defined in a CP directive.
RPL alias **RMTPLINK**	*RPL=name8:* Identifies the remote physical link defined in an RPL directive.
TP alias **TPROTOCOL**	*TP=name8:* Identifies the transport protocol defined in a TP directive.

6.11 RCT (RMTCTLR) Directive

Purpose

This directive applies only to the basic network configuration description. It describes a remote controller driving a remote ISL transceiver. The RCT name appears in the RPL directive describing the transceiver.

Syntax

```
{ RCT     }
{         }          NAME = name8
{ RMTCTLR }

[ { RMR     }   { 3000 } ]
[,{         } = {      } ]
[ { RMRESET }   { dec5 } ]

[  { RMT1  }   { 500  } ]
[, {       } = {      } ]
[  { RMACK }   { dec5 } ]

[  { RMN2    }   { 5    } ]
[, {         } = {      } ]
[  { RMRETRY }   { dec2 } ]

[             { 1 } ]
[, RMTLOAD =  {   } ]  ;
[             { 0 } ]
```

Parameter Description

NAME — *NAME=name8:* Identifies the Remote Controller.

RMR alias **RMRESET** — Remote Maintenance Reset timer: The maximum acknowledgement delay in hundredths of a second after which the local TNS declares the remote controller inaccessible and sends it a *reset* frame.

*RMR=dec5, **Default**:* 3000 (30 seconds).

RMT1 alias **RMACK** — Remote Maintenance SIM-ACK timer: The maximum delay in hundredths of a second after which an unacknowledged SIM command is to be retransmitted.

*RMT1=dec5, **Default**:* 500 (5 seconds).

RMN2 alias **RMRETRY**	Remote Maintenance Retry count: The maximum number of times a SIM command can be retransmitted. *rmt1 * rmn2* is the maximum delay for acknowledging the SIM after which the local TNS declares the remote controller inaccessible. *RMN2=dec2,* ***Default****:* 5.
RMTLOAD	**Applicable only if the controller is administered by the local TNS and FECM servers for which the parameter RCT in the RSYS directive must specify a value other than NO**. Otherwise it is meaningless. *RMTLOAD=1:* ***Default*** The controller accepts remote requests for automatic loading. *RMTLOAD=0:* The controller does not accept such requests.

6.12 RPL (RMTPLINK) Directive

Purpose

This directive applies only to the basic network configuration description. It describes the Remote Physical Link connecting the ISL of the remote transceiver to its remote controller.
Although the RPL is associated with a single RCT, it can be shared by several NRs.

Syntax

```
{ RPL      }
{          }          NAME = name8
{ RMTPLINK }

 { RCT     }
,{         } = name8
 { RMTCTLR }

         [ { G }  ]
,IADDR = [ {   }: ] xx-xx-xx-xx-xx-xx
         [ { L }  ]

 { CB    }
,{       } = name8 ;
 { CABLE }
```

Parameter Description

NAME — *NAME=name8:* Identifies the Remote Physical Link.

RCT alias **RMTCTLR** — *RCT=name8:* Identifies the corresponding remote ISL controller.

IADDR — Represents the address of the transceiver on the ISL cable.
IADDR=[{L|G}:]xx-xx-xx-xx-xx-xx, where:

L: means Local
G: means Global, ***Default***
indicates that the associated ISL address is network-wide and unique
08-00-38 is the BULL S.A. (DPS 7000 system only) reserved upper part and *xx* are hexadecimal digits.

CB alias **CABLE** — *CB=name8:* Identifies the external name of the ISL cable relaying the transceiver.

6.13 RSC (RMTSESS) Directive

Purpose

This directive applies only to the basic network configuration description for passthrough and neighbor systems. It applies to both the basic network configuration description and the directory configuration description for remote systems. It defines:

- the DSA address of a DSA remote system.
- the OSI address of a remote system accessed via the PID module by means of the PIDTSEL and NSAP parameters. When omitted a default PIDTSEL is assumed.

OSI Syntax Format 1: Passthrough

```
{ RSC      }
{          }        NAME=name4
{ RMTSESS }

        { PT            }
,TYPE = { PASSTHRU      }
        { PASS-THROUGH }

 { ADDR      }
,{           } = ISO
 { ADDRTYPE }

[ { PRTC     }                  ]
[,{          } = { SID | ISO } ]
[ { PROTOCOL }                  ]

[ { PID     }    { PID_SELECT } ]
[,{         } = {            } ]
[ { PIDTSEL }    { hexa64     } ]

,NSAP = ( hexa40 [ hexa40 ]... )

[ { SVR    }         ]
[,{        } = name8 ]
[ { SERVER }         ]

[ { RTS          }         ]
[,{              } = name8 ]
[ { RMTTRANSPORT }         ]

,SCID = dec3 : dec3 ;
```

OSI Syntax Format 2: Neighbor

```
{ RSC      }
{          }         NAME=name4
{ RMTSESS  }

        { NB         }
,TYPE = { NEIGHBOR   }
        { NEIGHBOUR  }

 { ADDR      }
,{           } = ISO
 { ADDRTYPE  }

[ { PRTC      }                 ]
[,{           } = { SID | ISO } ]
[ { PROTOCOL  }                 ]

[ { PID      }    { PID_SELECT } ]
[,{          } =  {            } ]
[ { PIDTSEL  }    { hexa64     } ]

,NSAP = ( hexa40 [ hexa40 ]... )

[ { RTS           }         ]
[,{               } = name8 ] ;
[ { RMTTRANSPORT  }         ]
```

OSI Syntax Format 3: Remote System

```
{ RSC      }
{          }          NAME = name4
{ RMTSESS  }

        { RMT     }
,TYPE = {         }
        { REMOTE  }

 { ADDR      }
,{           } = ISO
 { ADDRTYPE  }

[ { PRTC      }                 ]
[,{           } = { SID | ISO } ]
[ { PROTOCOL  }                 ]

[ { PID      }    { PID_SELECT } ]
[,{          } =  {            } ]
[ { PIDTSEL  }    { hexa64     } ]

,NSAP = ( hexa40 [ hexa40 ]... ) ;
```

DSA Syntax Format

```
{ RSC      }              { BCLS   }   [         { RMT }]
{          }       NAME = {        }   [,TYPE =  { PT  }]
{ RMTSESS  }              { name4  }   [         { NB  }]

[ { ADDR     }       ]
[,{          } = DSA ]
[ { ADDRTYPE }       ]

[ { PRTC     }   { DSA | DSA200 } ]
[,{          } = {              } ]
[ { PROTOCOL }   { ISO | SID    } ]

[ { SVR    }                       ]
[,{        } =( name8 [,name8...] ) ]
[ { SERVER }                       ]

[ { RTS          }                             ]
[,{              } = ( name8 [, name8 ... ] ) ]
[ { RMTTRANSPORT }                             ]

,SCID = dec-3 : dec-3 ;
```

Parameter Description

NAME

Identifies the Remote Session Control. It must be the same as the name of the corresponding RSYS directive.

NAME=BCLS: **Applicable only in DSA**. System-reserved name dealing with Network Security Service. See paragraph 6.8.1 "Network Security Control Service".

TYPE

Specifies the type of the system to which the current session layer object belongs. If omitted:

TYPE is that explicitly declared or assumed by default, in the associated RSYS directive appearing just before the RSC directive
otherwise, TYPE is determined by the SVR and RTS parameters.
TYPE specified explicitly in the RSC and RSYS directives relating to the same system must be the same. TYPE is associated with ADDRTYPE which defines if the ISO syntax format is to be conformed to or ignored.

TYPE=PT (alias PASSTHRU alias PASS_THROUGH):
Default:
if ADDRTYPE=ISO is assumed
and SVR specifies an FEPS occurrence.

TYPE=NB (alias NEIGHBOR alias NEIGHBOUR):
Default:
if ADDRTYPE=ISO is assumed
and RTS is specified
and SVR does not specify an FEPS occurrence.

TYPE=RMT (alias REMOTE): ***Default***:
either if ADDRTYPE=DSA is assumed
or if ADDRTYPE=ISO is assumed and both SVR and RTS are omitted.

ADDR alias **ADDRTYPE**

Indicates that session addressing and syntax conforms to either the ISO/DSA or the DSA standard as specified by TYPE.
When omitted the default addressing type is that explicitly declared or assumed by default for the RSYS directive appearing immediately before the current RSC directive.
An error is flagged if the addressing type in the RSYS and RSC directives of the same system are not identical.

ADDR=ISO: ***Default***, if PIDTSEL or NSAP is specified.

ADDR=DSA: ***Default***, if PIDTSEL and NSAP are both omitted.

PRTC alias **PROTOCOL**

Specifies that the protocol is ISO/DSA or DSA according to ADDR explicitly declared or assumed by default.

PRTC= ISO or SID, ***Default*** if ADDR=ISO.

PRTC= DSA or DSA200, ***Default*** if ADDR=DSA.

PID alias **PIDTSEL**

Applicable only to ISO Syntax Formats. Represents the TSELECTOR part in the OSI addressing of the remote ISO address.

PID=hexa64: Specifies the transport layer selector for the OSI addressing of the remote system PID.

PID=PID_SELECT: ***Default***. Specifies a default transport layer selector for the OSI addressing of the remote system PID. Generates the value "5049445F53454C454354" in ASCII.

NSAP

Applicable to and mandatory for ISO Syntax Formats.

NSAP=(hexa40,hexa40...): Lists up to 20 Network Service Access Points attached to the local system and identifying remote connections. No default.

SCID

Applies to DSA Syntax Formats and ISO Syntax Format 1. Specifies the DSA address of the remote session control.
Since the administration of the front-end processor is still based on DSA objects and addressing, SCID is mandatory with ISO Syntax Format 1. SCID, however, does not apply to other ISO Syntax Formats.
SCID=dec3:dec3, where 0 <= *dec3* <= 255. No default.

SVR alias **SERVER**

Applies to DSA Syntax Formats and ISO Syntax Format 1.
DSA - *SVR=(name8 [,...]):* Lists up to 9 communications servers, TNS, FEPS occurrences, and up to 2 OCS server occurrence names, through which the Remote Session can be accessed. See the SVR directive.

Note:
TNS is the default if a RTS list is present.
TNS must not be specified when the RTS list is missing.
If RSC name is BCLS, the listed SVR names must map onto only passthroughs over the PSI which are dedicated to Network Security Control servers.

ISO - *SVR=name8:*

Applicable only to ISO Syntax Format 1.
Mandatory parameter for a passthrough linked over a PSI channel to the local system. Specifies the name of the FEPS occurrence which handles the PSI.

Note: SVR and RTS cannot both be omitted.

RTS alias **RMTTRANSPORT**

Applies to ISO Syntax Formats 1 and 2 and DSA Syntax Formats.

DSA - *RTS=(name8 [,...]):* Lists up to 9 transport stations for accessing the remote session through TNS, namely:
8 remote transport stations
and one belonging to the system declared by the RSC which is attached to the ISL, to ensure the most direct communications access, see Note.

ISO - *RTS=name8:* Mandatory for a passthrough or neighbor attached to the same ISL as the local system and must specify the transport station belonging to the passthrough or neighbor.
Each RTS in the list is declared by an RTS directive.
This parameter is meaningless if the remote session is not accessed by TNS.

Note:
The session routing function distributes connections over all the RTSs in the list to try to balance oad (except if QUOTAs have been explicitly defined via the directive SR).
If the RSC name is BCLS, the list of RTS names must only contain the transport stations belonging to passthroughs connected over the ISL which are dedicated to Network Security Control servers. See paragraph 6.8.1.

NOTE:
SVR and RTS cannot both be omitted.

6.14 RSYS (RMTSYSTEM) Directive

Purpose

All syntax formats of this directive apply to the basic network configuration description. However, only syntax format 3 applies to the directory configuration description. It describes how a system is to dialog with the local system.

NOTE:

In the syntax for PROFILE, *if several releases apply for a system,* the release taken as default when its abbreviated profile is specified, is shown in ***bold italics***.

The allowed PROFILE depends on the DPS 7000 system model when the TYPE is Passthrough or Neighbor.

The default value of SERVER and RMTCTLR depend on the DPS 7000 system model.

Syntax Format 1: Passthrough

Syntax Format 1-1: Passthrough for DPS 7000

```
{ RSYS       }
{            }        NAME = name4
{ RMTSYSTEM }

[         { PT            } ]
[,TYPE =  { PASSTHRU      } ]
[         { PASS_THROUGH  } ]

[ { ADDR      }    { DSA } ]
[,{           } =  {     } ]
[ { ADDRTYPE  }    { ISO } ]

[ { PF       }    { DN7100 | 'DN7100/DN-C/{V3.0|V4.1}'} ]
[,{          } =  {                                   } ]
[ { PROFILE  }    { CNP7   | 'CNP7/CNS7/{A0|A1|A2}'   } ]

[ { { SVR     }    { NO                      } } ]
[ { {         } =  {                         } } ]
[ { { SERVER  }    { name8                   } } ]
[,{                                            } ]  ;
[ { { RCT      }    { NO                     } } ]
[ { {          } =  {                        } } ]
[ { { RMTCTLR  }    { name8 [, name8 ] ...   } } ]
```

Syntax Format 1-2: Passthrough for DPS 7000/XTA

```
{ RSYS       }
{            }           NAME = name4
{ RMTSYSTEM }

[          { PT           }]
[, TYPE  = { PASSTHRU     }]
[          { PASS THROUGH }]

[  { ADDR     }    { DSA }]
[, {          } =  {     }]
[  { ADDRTYPE}     { ISO }]

  { PF      }   {                                 }
, {         } = { MW2600LE | 'MW2600LE/DNS-E/V3' }
  { PROFILE }   {                                 }

[  { SVR      }    {                }]
[, {          } =  { NO             }]
[  { SERVER   }    {                }]

[  { RCT      }    {                }]
[, {          } =  { name8 [,name8] }];
[  { RCTCTLR  }    {                }]
```

Syntax Format 2: Neighbor

Syntax Format 2-1 : Neighbor system for DSP7000

```
{ RSYS       }
{            }          NAME = name4
{ RMTSYSTEM }

        { NB        }
,TYPE = { NEIGHBOR  }
        { NEIGHBOUR }

[ { ADDR      }    { DSA } ]
[,{           } =  {     } ]
[ { ADDRTYPE  }    { ISO } ]
```

```
[                 { DPS7    | 'DPS7/GCOS-7/{V9|V10}'     } ]
[                 { DN7100  | 'DN7100/DN-C/{V3.0|V4.1}'  } ]
[,{ PF       }    { CNP7    | 'CNP7/CNS7/{A0|A1|A2}'     } ]
[ {          } =  { STID    | 'STID/ISO/{SID3|SID4}'     } ]
[ { PROFILE }     { NIP     |                            } ]
[                 { DPS6    | 'DPS6/DSS/{3.1|4.0|4.1}'   } ]
[                 { DPS6P   | 'DPS6P/HVS/{1.0|2.0}'      } ]
[                 { MW2600  | 'MW2600/DNS-E/V3'          } ]

[ { RCT      }    { NO                       } ]
[,{          } =  {                          } ] ;
[ { RMTCTLR }     { name8 [, name8 ] ...     } ]
```

Syntax Format 2-2 : Neighbor system for DSP7000/XTA

```
{ RSYS        }
{             }        NAME = name4
{ RMTSYSTEM }

        { NB          }
,TYPE = { NEIGHBOR    }
        { NEIGHBOUR }

[ { ADDR      }    { DSA } ]
[,{           } =  {     } ]
[ { ADDRTYPE }     { ISO } ]

 { PF       }
,{          } =  { MW2600LE | 'MW2600LE/DNS-E/V3' }
 { PROFILE }

[ { RCT      }    { NO                       } ]
[,{          } =  {                          } ] ;
[ { RMTCTLR }     { name8 [, name8 ] ...     } ]
```

Syntax Format 3: Remote System

```
{ RSYS      }                      { BCLS  }
{           }          NAME = {       }
{ RMTSYSTEM }                      { name4 }

[         { RMT    } ]
[,TYPE = {        } ]
[         { REMOTE } ]

[ { ADDR     }   { DSA } ]
[,{          } = {     } ]
[ { ADDRTYPE }   { ISO } ]

[               { DPS7     |'DPS7/GCOS-7/{V9|V10}'                 }]
[               { DPS6     |'DPS6/DSS/{2.2|3.0|3.1|4.0|4.1}'       }]
[               { DPS6P    |'DPS6P/HVS/{1.0|2.0}'                  }]
[               { DN7100   |'DN7100/DN-C/{V3.0|V4.1}'              }]
[ { PF      }   {           'DN7100/DN-B/{V2.5|V2.6}'              }];
[,{         } = { DPS8     |'DPS8/GCOS-8/SR{2500|3000|4000|5000}'}]
[ { PROFILE }   { DPS4     |'DPS4/GCOS-4/{NCT1|NCT2}'              }]
[               { CNP7     |'CNP7/CNS7/{A0|A1|A2}'                 }]
[               { M400     |'M400/DSA/{3.0|3.1|4.0}'               }]
[               { STID     |'STID/ISO/{SID3|SID4}'                 }]
[               { IBM      |'IBM/JANUS/DN300'                      }]
[               { MW2600   |'MW2600/DNS-E/V3'                      }]
[               { MW2600LE|'MW2600LE/DNS-E/V3'                    }]
```

Parameter Description

NAME

NAME=name4: Identifies the network-wide unique name of the system. It is also the name of its associated RSC.

NAME=BCLS: **Applicable only to Remote Systems Format 3**.
When the local system is declared *secured* in the basic network configuration description, the system-reserved name BCLS represents a dummy remote system used by the Network Security Control Service. See paragraph 6.8.1.

Restriction when using QMON:
If QUEUES are declared, this name must be different from that specified in the NETWORK directive.

TYPE Defines the type of the system in relation to the local system:

TYPE=PT (alias PASSTHRU alias PASS_THROUGH)
Identifies the system as a passthrough:
either over the PSI if SVR parameter is supplied
or over the ISL if RCT parameter is given.
Such a system can be specified as belonging to an ISO addressing subdomain. See SDOM directive.

TYPE=NB (alias NEIGHBOR alias NEIGHBOUR)
Identifies the system as a neighbor attached to the same ISL cable as the local system but not used by the local system as a passthrough. Such a system can be specified as belonging to an ISO addressing subdomain. See SDOM directive.

TYPE=RMT (alias REMOTE)
Identifies the system as a remote system when both SVR and RCT parameters are omitted. Such a system can be accessed by the local system only by means of a passthrough.

The default depends on the profile.

ADDR alias **ADDRTYPE** Indicates that the addressing used to access the current system is OSI. When omitted, the default addressing type is:

either that explicitly declared or assumed by default for the associated RSC directive
or DSA.

An error condition is flagged if the addressing type differs between the RSYS and RSC directives describing the same system.

ADDR=DSA: ***Default***
ADDR=ISO

PF alias **PROFILE**

Identifies the system's hardware type, operating system and the software release which determine the capabilities of its DJP and UFT applications with the local DPS 7000.

PF=brief: Brief profile specifying only the hardware type. In this case the capabilities assumed are those of the most recent operating system release for this system.

Defaults:
DN7100 if SVR parameter is specified
CNP7 if RCT parameter is specified
DPS7 if both SVR and RCT are omitted.
PF='extended': Extended profile enclosed between single quotes.

DPS 7000 FILE TRANSFER CAPABILITIES
The file transfer capability of a remote DPS 7000 towards the local DPS 7000 system is security for the file transfer acceptor DPS 7000 (local or remote):

- if it is locally configured as *protected* with respect to the file transfer application, the file transfer will run only after agreement by this system of the submitter's access rights.
- if it is locally configured as *not protected* with respect to the file transfer application, the transfer will run normally.

DPS 7000/XTA:

MW2600LE/DNS-E/V3 is applicable to remote systems.

It is applicable to Passthrough or Neighbor systems with keyword SVR=NO and RCT=name8 only for administration MainWay 2600LE.

SVR alias **SERVER**

Applicable only to passthroughs linked to the local system through a PSI channel.

SVR=name8: Identifies the FEPS occurrence in the local DPS 7000 administering this passthrough. *SVR=name8* can only be specified with *RCT=NO*.
SVR=NO: Indicates that the system is a passthrough and is not administered by the local DPS 7000. It is the default value when:
TYPE=PASS_THROUGH is specified
both SVR and RCT are omitted
and PROFILE is either DN7100 or omitted.

RCT alias **RMTCTLR**

Applicable only to passthrough or neighbors linked to the local DPS 7000 through the ISL.

RCT=(name8...): Lists up to 4 remote controllers through which this passthrough or neighbor is administered by TNS of the local DPS 7000 through LOAD, DUMP, and SYSGEN commands addressed to the local FECM server. *RCT=(name8...)* can only be specified with *SERVER=NO*.
RCT=NO: Indicates that the system is a passthrough or neighbor and is ***not*** administered locally by the DPS 7000. It is the default for a
neighbor with a profile DN7100 or CNP 7
passthrough with profile CNP7.

DPS 7000/XTA : It is the default for a
Neighbor and Passthrough with a profile MW2600LE.

DPS 7000: *SVR and RCT are not applicable to remote systems or neighbor systems with profiles other than DN7100 or CNP7 .*

DPS 7000/XTA: *SVR is not applicable to remote, passthrough or neighbor systems.*

RCT is not applicable to remote systems.

6.15 RTS (RMTTRANSPORT) Directive

Purpose

This directive applies only to the basic configuration description. It describes a remote ISL transport station located on a neighbor or passthrough.

Syntax

```
{ RTS          }
{              }          NAME = name8
{ RMTTRANSPORT }

 { NR       }
,{          } = name8  [ ,name8 ... ]
 { NETROUTE }

[          { 2200 } ]
[,MAXCN = {      } ]
[          { dec4 } ]

[ { N                 }   { 3    } ]
[,{                   } = {      } ]
[ { RETRY | MAXRETRY  }   { dec2 } ]

[ { T1                  }   { 10   } ]
[,{                     } = {      } ]
[ { FLOW | RETRYDELAY   }   { dec2 } ]

[ { T2                   }        ]
[,{                      } = dec3 ]
[ { SILENT | IDLEDELAY   }        ]

[ { TP        }   { ISO[:1] } ]
[,{           } = {         } ] ;
[ { TPROTOCOL }   { ISO:2   } ]
```

Parameter Description

NAME	*NAME=name8:* Identifies the Remote Transport Station.
NR alias **NETROUTE**	*NR=(name8 ...):* Lists up to 16 NRs for accessing the RTS, each NR being defined by an individual NR directive.
MAXCN	Defines the maximum number of transport connection that may be established between the local system and the remote transport station. This parameter does not affect the consumption of resources. Only active transport connections consume resources. 1 <= *maxcn* <= 2200. ***Default***: 2200.

T1 or **FLOW** or **RETRYDELAY**

Timeout in seconds before retransmission of a TPDU if no ACK is received.

1 <= *flow* <= 30. ***Default***: 10 seconds.
When FLOW, SILENT, RETRY are all omitted, the following selected defaults are compatible with those of the CNP 7:

FLOW=10, SILENT=60, RETRY=3.

T2 or **SILENT** or **IDLEDELAY**

Timeout in seconds before the connection is released, because no TPDU has been received (protection against unsignalled failures).

1 <= *silent* <= 600. ***Default***: (2 * *retry* * *flow*) seconds.

When FLOW, SILENT, RETRY are all omitted, the following selected defaults are compatible with those of the CNP 7:

FLOW=10, SILENT=60, RETRY=3.

N or **RETRY** alias **MAXRETRY**

Maximum number of times the local transport is to retransmit a TPDU which should be have been ACKed but was not.
1 <= *retry* <= 10. ***Default****:* 3.

When FLOW, SILENT, RETRY are all omitted, the following selected defaults are compatible with those of the CNP 7:

FLOW=10, SILENT=60, RETRY=3.

TP alias **TPROTOCOL** Specifies the transport protocol type and its version.

TP=ISO:1 or ISO - Transport layer server supports SID addressing where the protocol is ISO and the address form is ISO-DSA.

TP=ISO:2. Transport layer server supports full ISO addressing and protocol.

Default*:* TP=ISO:1.

6.16 SR (SESSROUTE) Directive

Purpose

This directive applies to both the basic network and directory configuration descriptions. It is optional and allows the local session control to select the local server and, when TNS, the remote transport station by which it can access a remote session control.

The session route must always be described in the same configuration description (basic network or directory) that the session control to which it is attached.

When omitted, session routing is automatically generated with default values:

- for each server specified in the SVR parameter of the RSC directive
- for each transport station specified in the RTS parameter of the RSC directive.

Where no SR has been specified at all, all inward and outward connections for all RSCs configured will be equally shared among all available routes.

Since the SR describes a network path and not an object in a layer, it does not have a name but is instead identified by its two endpoints.

NOTE:
The default value of STATE and QUOTA for SVR TNS depends on the DPS 7000 system model.

Syntax Format 1: For SVR FEPS or OCS

```
{ SR        }
{           }
{ SESSROUTE }

 { RSC      }
,{          } = name4
 { RMTSESS  }

 { SVR    }
,{        } = name8
 { SERVER }

[ { ST    }   { ENBL } ]
[,{       } = {      } ]
[ { STATE }   { LOCK } ]

[          { 1    } ]
[,QUOTA =  {      } ]  ;
[          { dec2 } ]
```

Syntax Format 2: For SVR TNS

Syntax Format 2-1 : For SVR TNS (for DPS 7000)

```
{ SR         }
{            }
{ SESSROUTE  }

 { RSC      }
,{          } = name4
 { RMTSESS  }

[ { SVR     }       ]
[,{         } = TNS ]
[ { SERVER  }       ]

 { RTS            }
,{                } = name8
 { RMTTRANSPORT   }

[ { ST     }   { ENBL  } ]
[,{        } = {       } ]
[ { STATE  }   { LOCK  } ]

[          { 1    } ]
[,QUOTA =  {      } ]  ;
[          { dec2 } ]
```

Syntax Format 2-2 : For SVR TNS (for DPS 7000/XTA)

```
{ SR         }
{            }
{ SESSROUTE  }

 { RSC      }
,{          } = name4
 { RMTSESS  }

 { SVR     }
,{         } = TNS
 { SERVER  }

 { RTS            }
,{                } = name8
 { RMTTRANSPORT   }

[ { ST     }   {       } ]
[,{        } = { LOCK  } ]
[ { STATE  }   {       } ]

[          {      } ]
[,QUOTA =  {  0   } ]  ;
[          {      } ]
```

Parameter Description

RSC alias **RMTSESS** — *RSC=name4:* Identifies the remote session control whose objects are to be accessed. A given SR must be declared in the basic network or directory configuration description, if its corresponding session control is described.

SVR alias SERVER — **Specifies the communications server:**

SVR=name8: **Applicable only to Format 1. Specifies the name of a FEPS or OCS occurrence in the local system used to access the remote session layer object**.
Default is *SVR=TNS* if RTS is specified.

RTS alias **RMTTRANSPORT**

RTS=name-8: **Applicable only to Format 2.** The session routing applies to the remote transport station and is used by TNS to access the remote session layer object.

ST alias **STATE** — Defines the initial state of the SR.

ST=ENBL: ***Default*** for DPS 7000, the route is selectable.

ST=LOCK: ***Default*** for DPS 7000/XTA, the route is not selectable.

QUOTA — Meaningful only when more than one SR is available to access the remote session layer object.

QUOTA=dec2, where 0 <= *quota* <= 32, defines for this SR the weighting ratio used by the alternative dynamic session routing to select an SR.
All inward and outward connections to a given remote session control will be shared among its available and valid SRs according to the quota specified for each one.

QUOTA=1 : Default for DPS 7000, allows using alternative dynamic session routing.

QUOTA=0 : Default for DPS 7000/XTA, does not allow using alternative dynamic session routing.

EXAMPLE:

Three SRs (SR1, SR2 and SR3) are available to access the same Session Control. SR1 is defined with QUOTA=5, SR2 with QUOTA=2, and SR3 with QUOTA=3. If 10 connections are requested to the Session Control, the routing ratio for the SRs will be in the weighting proportion of their QUOTAs, namely, 5, 2 and 3, respectively for SR1, SR2 and SR3.

❑

6.17 SVR (SERVER) Directive

Purpose

This directive applies only to the basic network configuration description. It describes a communications server for handling either the Datanet, the CNP 7, or the OCS Front End. The server can be one FEPS occurrence, one OCS occurrence, or TNS. Whereas up to 8 FEPS occurrences can be declared, only one SVR directive for TNS is allowed. Up to 4 OCS server occurrences can be declared.

DPS 7000/XTA: TNS Server is used only for MainWay 2600LE administration.

It is reserved to establish administrative connections.

Syntax Format 1: FEPS

```
{ SVR    }
{        }          ,NAME = name8
{ SERVER }


,TYPE = FEPS


 { LCT     }                           }
,{         } = ( CChexa2 [,CChexa2 ] ) }
 { LOCCTLR }                           }


[ { BUFSZ   }   { 128  } ]
[,{         } = {      } ]
[ { BUFSIZE }   { dec4 } ]


[,BUFNB = (dec6, dec6) ]


[ { EXT       }   { 80   } ]
[,{           } = {      } ] ;
[ { EXTENSION }   { dec3 } ]
```

Syntax Format 2: TNS

```
{ SVR    }
{        }           [,NAME = TNS ]
{ SERVER }

[, TYPE = TNS ]

[ { BUFSZ   }   { 128  } ]
[,{         } = {      } ]
[ { BUFSIZE }   { dec4 } ]

[,BUFNB = (dec6, dec6) ]

[ { EXT       }   { 80   } ]
[,{           } = {      } ] ;
[ { EXTENSION }   { dec3 } ]
```

Syntax Format 3: OCS

```
{ SVR    }
{        }          ,NAME = name-8
{ SERVER }

,TYPE = OCS

 {LCT    }
,{       } = EAhexa2
 {LOCCTLR}

[ { BUFSZ   }   { 128  } ]
[,{         } = {      } ]
[ { BUFSIZE }   { dec4 } ]

[,BUFNB = (dec-6, dec-6) ]

[ { EXT       }   { 80   } ]
[,{           } = {      } ] ;
[ { EXTENSION }   { dec3 } ]
```

Parameter	**Description**
NAME	Identifies the communications server handling the FEP. In the case of OCS, identifies the communications server handling the front end. *NAME=name8:* **Applicable only to Format 1.** Mandatory parameter identifying an FEPS occurrence, for which *TYPE* ***must*** be FEPS. Up to 8 FEPS occurrences can be declared, each by a separate SVR directive. *NAME=TNS:* ***Default*** **- Applicable only to Format 2.** Only one occurrence of TNS is allowed.
TYPE	Specifies which type of server applies. *TYPE=TNS:* ***Default*** **- Applicable only to Format 2**. *TYPE=FEPS:* **Applicable only to Format 1**. *TYPE=OCS:* **Applicable only to Format 3**.
LCT alias **LOCCTLR**	**For Format 1** lists up to 2 local controllers. *LCT=(CChexa2 [,CChexa2]), where CChexa2 identifies each LCT.* **Note:** If enabling or saving is requested in the CRNETGEN command, this name is searched in the SRST specified by the SRSTFILE parameter of the CRNETGEN command. It identifies the PSI channel linking the Datanet passthrough, and the FEPS occurrence managing the Datanet. See Section 2. **For Format 3** (OCS), gives the name of the front end driven by the OCS occurrence. The name must be the same as the name of the adaptor or transceiver given in the SRST table (external identification EAhexa2). Up to 4 OCS servers can be declared (each by a separate SVR directive).

7. Improving the Network Description

When the network configuration is described by high-level SYSTEM and DEFAULT directives, the resulting generation is satisfactory for most networks. See Section 3.

However, complex network configurations require more detailed and precise arguments which only low-level directives can supply.

The rules described in this Section must be rigorously adhered to when mixing SYSTEM directives with low-level LSYS/LSC and RSYS/RSC directives.

For detailed network configuration description improvement, the user may get help from the low-level description using one of following features:

Expansion Overriding

Expansion overriding involves inserting low-level directives in front of the SYSTEM directive concerned to inhibit and so override an attribute or attributes which would otherwise be supplied in its usual expansion.

Default Overriding

Default overriding involves using the DEFAULT directive which is remanent and can only alter some attribute(s) when subsequent SYSTEM directives are expanded.

Adding Insertion

Adding insertion involves freely inserting anywhere in the network configuration description, low-level directives which do not affect the expansion of SYSTEM directives.

7.1 Expansion overriding

To use expansion overriding, the user must know:
- how the SYSTEM directive expands and what its result is
- what convention is used for naming objects
- and how the expansion overriding mechanism works.

7.1.1 Expanding the SYSTEM Directive

Each SYSTEM directive internally expands into a set of low-level directives whose semantics is the strict equivalent of the SYSTEM directive itself.

The complete expansion of the SYSTEM directive can be displayed in the NETGEN report by specifying OBJLIST=1 in the SYSTEM directive itself:

- each resulting low-level directive appears immediately after its source SYSTEM directive, indented to the right and preceded by ->
- the SYSTEM directive itself also appears to show the default parameters which are taken into account.

This Section deals with the detailed expansion of the SYSTEM directive for each type of system, as follows:

- A general example of its SYSTEM directive, for which each subsequent line:
  - when headed by *, gives the principal conditions of the expansions
  - when indented, denotes secondary conditions.
- The whole expansion is given in the sequence as it actually appears, which the following conventions:
  - parts underlined in the resulting low-level directives come directly or after computing, from names or arguments also underlined in the corresponding SYSTEM directive
  - the outline of each resulting low-level directive starts with -> and ends with ;
  - parameters actually expanded but deliberately omitted are indicated with ...
  - only the parameters directly taken from the SYSTEM directive are shown.
- The expansions shown assume that neither expansion overriding nor default overriding has occurred.
- The four kinds of expansions on the local system, passthrough, neighbor and remote system, are mutually consistent in that some values such as PSI names, local ISL number and cable names, given for the local system are used in the expansion of other systems.

7.1.1.1 Expansion of Local System

ISO addressing for DPS 7000

```
SYS  host TYPE=LOC SCID=12:34 PID=hexa1
        [ TSEL=hexa2 NSAP=(hexa3 [, hexa4] ... ) ]
        [ ISL1=(12-34-56,EA11)         ]
        [ ISL2=(23-45-67,EA12)         ]
        [ ISL3=(34-56-78,EA13,CBL2)    ]
        [ ISL4=(45-67-89,EA14,CBL3)    ]
        [ NCC1=(98-76-54,EA21)         ]
        [ NCC2=(87-65-43,EA22)         ]
        [ NCC3=(76-54-32,EA23)         ]
        [ NCC4=(65-43-21,EA24)         ]
        [ POOLSIZE=(2,5) ] [ TCPIP     ] ;
```

DSA addressing for DPS 7000

```
SYS  host TYPE=LOC SCID=12:34
        [ ISL1=(12-34-56,EA11) ]
        [ ISL2=(23-45-67,EA12) ]
        [ ISL3=(34-56-78,EA13,CBL2) ]
        [ ISL4=(45-67-89,EA14,CBL3) ]
        [ NCC1=(98-76-54,EA21)         ]
        [ NCC2=(87-65-43,EA22)         ]
        [ NCC3=(76-54-32,EA23)         ]
        [ NCC4=(65-43-21,EA24)         ]
        [ POOLSIZE=(2,5) ]
        [ PT = ( psys [, psys ] ... ) ] ;
```

ISO addressing for DPS 7000/XTA

```
SYS host TYPE=LOC SCID=85:05 PID=hex1
         [ TSEL=hexa2 NSAP=(hexa3 [,hexa4]…)   ]
         [ ISL1=(00-90-27-01-30-50, EA21, CBL1)]
         [ NCC1=(00-90-27-01-30-51, EA22)      ]
         [ POOLSIZE=(2,5)                      ] ;
```

DSA addressing for DPS 7000/XTA

```
SYS host TYPE=LOC SCID=85:05
             [ ISL1=(00-90-27-01-30-50, EA21, CBL1)]
             [ NCC1=(00-90-27-01-30-51, EA22)      ];
```

```
Expansion of SYSTEM Directive for the Local System:
```

* If the NETWORK directive has not already been specified:

```
-> NETWORK host;
```

* The LSYS directive is always generated:

```
-> LSYS host PF=DPS7 LSC=host
```

- If PID=ISO or hex1 has been specified:
  `ADDR=ISO;`
- If PID=DSA or SID has not been specified:
  `ADDR=DSA;`

* The LSC directive is always generated:

```
-> LSC  host SCID=12:34;
```

- If PID=DSA has not been specified:
  `ADDR=DSA PRTC=DSA200`
- If PID=SID has been specified:
  `ADDR=DSA PRTC=SID`
- If PID=ISO has been specified:
  `ADDR=ISO PRTC=SID PID=PID_SELECT`
- If PID=hex1 has been specified:
  `ADDR=ISO PRTC=SID PID=hex1`
- If TSEL=hex2 has been specified:
  `TSEL=hex2`
- If NSAP=hex3 has been specified:
  `NSAP=(hex3 [, hex4] ... );`

* If ISL1 has been specified:

```
-> SVR  TNS TYPE=TNS
```

- If POOLSIZE has been specified:

```
[ BUFSIZE=128,BUFNB=(1024,2560) ] ... ;
-> CP   CP_EA11 LPL=LPL_EA11 ...;
-> LPL  LPL_EA11 CB=CBL_MAIN IADDR=08-00-38-12-34-56
   LCT=EA11 ... ;
-> LCT  EA11 TYPE=ISL ...
```

* If MADDR has been specified:

```
[ MADDR=(G:xx-xx-xx-xx-xx-xx,G:xx-xx-xx-xx-xx-xx) ]
->  CB   CP_EA11 ...;
->  LPL  LPL_EA11,CB=CBL_MAIN,IADDR=08-00-38-12-34-56
         MADDR=(G:xx-xx-xx-xx-xx-xx,G:xx-xx-xx-xx-xx-xx);
->  LCT  EA11 ...;
```

* If RADDR has been specified:

```
     [ RADDR=(xd-xx-xx-xx-xx-xx,xd-xx-xx-xx-xx-xx) ]
     ->  CB   CP_EA11 ...;
     ->  LPL  LPL_EA11,CB=CBL_MAIN,IADDR=08-00-38-12-34-56
              RADDR=(xd-xx-xx-xx-xx-xx,xd-xx-xx-xx-xx-xx);
     ->  LCT  EA11 ...;
```

* If MADDR and RADDR have been specified:

```
     [ MADDR=(G:xx-xx-xx-xx-xx-xx,G:xx-xx-xx-xx-xx-xx),
       RADDR=(xd-xx-xx-xx-xx-xx,xd-xx-xx-xx-xx-xx) ]
     ->  CB   CP_EA11 ...;
     ->  LPL  LPL_EA11,CB=CBL_MAIN,IADDR=08-00-38-12-34-56
              MADDR=(G:xx-xx-xx-xx-xx-xx,G:xx-xx-xx-xx-xx-xx)
              RADDR=(xd-xx-xx-xx-xx-xx,xd-xx-xx-xx-xx-xx);
     ->  LCT  EA11 ...;
```

* If TCPIP has been specified:

```
     [ TCPIP=(IP,ARP,RARP,TRAIL0,TRAIL1,TRAIL2,TRAIL3)]
     [ LT=(ISO1EA11,ISO2EA11,TCP1EA11,TCP2EA11,RMLTEA11)];
     ->  LT   ISO1EA11 LSAP=ISO;
     ->  LT   ISO2EA11 LSAP=ISO;
     ->  LT   TCP1EA11 LSAP=TCPIP MAXCCE=(1,6);
     ->  LT   RMLTEA11 LSAP=RM;
```

* If ISL2 has been specified:

```
  -> CP   CP_EA12 LPL=LPL_EA12 ...;
  -> LPL  LPL_EA12 CB=CBL_MAIN IADDR=08-00-38-23-45-67
     LCT=EA12 ...;
  -> LCT  EA12 TYPE=ISL ...
```

- If TCPIP has been specified:

```
     [ TCPIP=(IP,ARP,RARP,TRAIL0,TRAIL1,TRAIL2,TRAIL3)]
     [ LT=(ISO1EA12,ISO2EA12,TCP1EA12,TCP2EA12,RMLTEA12)];
     ->  LT   ISO1EA12 LSAP=ISO;
     ->  LT   ISO2EA12 LSAP=ISO;
     ->  LT   TCP1EA12 LSAP=TCPIP MAXCCE=(1,6);
     ->  LT   RMLTEA12 LSAP=RM;
```

* If ISL3 has been specified:

```
  -> CP   CP_EA13 LPL=LPL_EA11 ...;
  -> LPL  LPL_EA13 CB=CBL2 08-00-38-34-56-78 LCT=EA13 ... ;
  -> LCT  EA13 TYPE=ISL ...
```

- If TCPIP has been specified:

```
     [ TCPIP=(IP,ARP,RARP,TRAIL0,TRAIL1,TRAIL2,TRAIL3) ]
     [ LT=(ISO1EA13,ISO2EA13,TCP1EA13,TCP2EA13,RMLTEA13) ];
     ->  LT   ISO1EA13 LSAP=ISO;
     ->  LT   ISO2EA13 LSAP=ISO;
     ->  LT   TCP1EA13 LSAP=TCPIP MAXCCE=(1,6);
     ->  LT   RMLTEA13 LSAP=RM;
```

* If ISL4 has been specified:
```
-> CP   CP_EA14 LPL=LPL_EA14 ...;
-> LPL  LPL_EA14 CB=CBL3 08-00-38-45-67-89 LCT=EA14 ... ;
-> LCT  EA14 TYPE=ISL ...
```

- If TCPIP has been specified:
```
[ TCPIP=(IP,ARP,RARP,TRAIL0,TRAIL1,TRAIL2,TRAIL3) ]
[ LT=(ISO1EA14,ISO2EA14,TCP1EA14,TCP2EA14,RMLTEA14) ];
->  LT   ISO1EA14 LSAP=ISO;
->  LT   ISO2EA14 LSAP=ISO;
->  LT   TCP1EA14 LSAP=TCPIP MAXCCE=(1,6);
->  LT   RMLTEA14 LSAP=RM;
```

When the ENET directive is encountered or at the end of the basic network description:

* If TCPIP has been specified and a local CP rank 3 remains unused:
```
**********************************************************
* The local system using TCP/IP is configured alone on   *
* the ISL cable CBL2 and CP CP_EA13, and the X__3  dummy *
* CNP7 is generated for internal consistency.            *
**********************************************************
->  SYS  X__3 PF=CNP7 SCID=255:253 ISL=5F-FF-F3 CBL2);
->  RSYS NAME=X__3 TYPE=PT PF=CNP7 RSC=CNP0 RCT=RCT1X__3;
->  RSC  NAME=X__3 SCID=255:253 SVR=TNS RTS=RTS_X__3;
->  RTS  NAME=RTS_X__3 NR=NR31X__3 TP=ISO:02;
->  NR   NAME=NR31X__3 CP=CP_EA13 RPL=RPL1X__3 TP=TP_DEFLT;
->  RPL  NAME=RPL1X__3 CB=CBL_MAIN RCT=RCT1X__3
         IADDR=08-00-38-5F-FF-F3;
->  RCT  NAME=RCT1X__3;
```

* If not yet generated:
```
->  TP   NAME=TP_DELFT;
```

* If PT has been specified:
```
***************************************************************
* The local system being declared secured the following dummy *
* remote DPS7 is generated for local connections security     *
***************************************************************
->  SYS  BCLS TYPE=RMT PF=DPS7 SCID=12:34 PT=(psys [, psys ]...)
                   PID=DSA ;
->  RSYS NAME=BCLS TYPE=RMT ADDR=DSA PF=DPS7 RSC=BCLS ;
->  RSC  NAME=BCLS TYPE=RMT ADDR=DSA PRTC=DSA200 SCID=12:34
                   SVR=(SVR_psys [,SVR_psys ] ... );
```

* If NCCx is specified, the expansion is:
```
LCT NAME = EAxx  TYPE = NCC  IADDR = 08-00-38-dh-hh-hh
SVR NAME = OCS_EAxx  TYPE = OCS  LCT = EAxx
```

* If POOLSIZE is specified:
```
BUFSIZE = 128  BUFNB = (256, 512)
```

7.1.1.2 Expansion of Passthrough

ISO Addressing

```
SYS psys TYPE=PT PF=DN7100  SCID=23:45 PID=hex1
          [ NSAP=(hex2 [,hex3] ... ) ]
          [ PSI=(CC01 [,CC02] ) ]
          [ ISL1=(56-78-9A,CBL2) ]
          [ ISL2=(67-89-AB) ]
          [ ISL3=(78-9A-BC,CBL_MAIN) ]
          [ ISL4=(89-AB-CD,CBL3) ]
          [ POOLSIZE=(5,10) ]  [ LOAD ] ;
```

DSA Addressing

```
SYS psys TYPE=PT PF=DN7100  SCID=23:45
        [ PSI=(CC01 [,CC02] ) ]
        [ ISL1=(56-78-9A,CBL2) ]
        [ ISL2=(67-89-AB) ]
        [ ISL3=(78-9A-BC,CBL_MAIN) ]
        [ ISL4=(89-AB-CD,CBL3) ]
        [ POOLSIZE=(5,10) ]  [ LOAD ]
        [ PT = ( fsys [, fsys ] ... ) ] ;
```

Expansion of SYSTEM Directive for Passthrough:

* The RSYS directive is always generated:
`-> RSYS psys TYPE=PT PF=DN7100 RSC=psys ...`

- If PID=ISO or hex1 has been specified :
  `ADDR=ISO`
- If PID=DSA or SID has not been specified :
  `ADDR=DSA`
- If PSI has been specified and LOAD=0 :
  `SVR=NO;`
- If PSI has been specified and LOAD=1 :
  `SVR=SVR_psys;`
- If PSI has not been specified, but ISL1 is, and LOAD=0 :
  `RCT=NO;`

- If PSI is has not been specified, but ISLs are, and LOAD=1 :
```
      RCT=(RCT1psys,RCT2psys,RCT3psys,RCT4psys);
```

* The RSC directive is always generated:
```
-> RSC  psys SCID=23:45
```

- If PID=DSA has not been specified:
```
      ADDR=DSA PRTC=DSA200
```

- If PID=SID has been specified:
```
      ADDR=DSA PRTC=SID
```

- If PID=ISO has been specified:
```
      ADDR=ISO PRTC=SID PID=PID_SELECT
```

- If PID=hex1 has been specified:
```
      ADDR=ISO PRTC=SID PID=hex1
```

- If NSAP=(hex2 [, hex3] ...) has been specified:
```
      NSAP=(hex2 [, hex3] ... )
```

- If PSI has been specified:
```
      SVR=(SVR_psys)
```

- If ISL1 has been specified:
```
      SVR=( ... , TNS)
      RTS=(RTS_psys)
```

- If PT has been specified: for each fsys over a PSI:
```
      SVR=( ... SVR_fsys)
```

- If PT has been specified: for each fsys over an ISL:
```
      RTS=( ... RTS_fsys);
```

* If PSI has been specified:
```
-> SVR  SVR_psys TYPE=FEPS LCT=(CC01 [,CC02]) ...
```

- If POOLSIZE has been specified:
```
   BUFSIZE=128,BUFNB=(2560,5120) ;
-> LCT  CC01 TYPE=PSI;
->[LCT  CC02 TYPE=PSI];
```

* If ISL1 has been specified:
```
-> RTS  RTS_psys NR=(NR31EA13 ...
```

- If ISL2 has been specified:
```
      ... NR32EA13 ...
```

- If ISL3 has been specified:
```
      ... NR13EA11,NR23EA12 ...
```

- If ISL4 has been specified:
```
... NR44EA14);
```
- If PF={ CNP7 | DN7100 }
  or PF >= { CNP7/CNS/A2 | DN7100/DN-C/V4.0 }:
```
TP=ISO:2 ;
```

* If ISL1 has been specified:
```
-> NR   NR31psys CP=CP_EA13 RPL=RPL1psys TP=TP_DEFLT;
-> RPL  RPL1psys CB=CBL2 RCT=RCT1psys IADDR=08-00-38-56-78-9A;
-> RCT  RCT1psys ;
```

- If the TP directive has not yet been generated:
```
-> TP   TP_DEFLT ;
```

* If ISL2 is specified:
```
-> NR   NR32psys CP=CP_EA13 RPL=RPL2psys TP=TP_DEFLT;
-> RPL  RPL2psys CB=CBL2 RCT=RCT2psys IADDR=08-00-38-67-89-AB;
-> RCT  RCT2psys ;
```

* If ISL3 is specified:
```
-> NR   NR13psys CP=CP_EA11 RPL=RPL3psys TP=TP_DEFLT;
-> NR   NR23psys CP=CP_EA12 RPL=RPL3psys TP=TP_DEFLT;
-> RPL  RPL3psys CB=CBL_MAIN RCT=RCT3psys
        IADDR=08-00-38-78-9A-BC;
-> RCT  RCT3psys ;
```

* If ISL4 is specified:
```
-> NR   NR44psys CP=CP_EA14 RPL=RPL4psys TP=TP_DEFLT;
-> RPL  RPL4psys CB=CBL3 RCT=RCT4psys IADDR=08-00-38-89-AB-CD;
-> RCT  RCT4psys ;
```

7.1.1.3 Expansion of Neighbor

ISO Addressing

```
SYS nsys TYPE=NB PF=DPS7 PID=hex1
         [ NSAP=(hex2 [,hex3] ... ) ]
           ISL1=(9A-BC-DE,CBL3)
         [ ISL2=(AB-CD-EF) ]
         [ ISL3=(BC-DE-F0,CBL2) ]
         [ ISL4=(CD-EF-01) ]
         [ LOAD ] ;
```

DSA Addressing

```
SYS nsys TYPE=NB PF=DPS7 SCID=34:56
           ISL1=(9A-BC-DE,CBL3)
         [ ISL2=(AB-CD-EF) ]
         [ ISL3=(BC-DE-F0,CBL2) ]
         [ ISL4=(CD-EF-01) ]
         [ LOAD ]
         [ PT = ( fsys [, fsys ] ... ) ] ;
```

Expansion of SYSTEM Directive for Neighbor:

* The RSYS directive is always generated:
`-> RSYS nsys TYPE=NB PF=DPS7 RSC=nsys ...`

- If PID=ISO or hex1 has been specified:
  `ADDR=ISO`

- If PID=DSA or SID has not been specified:
  `ADDR=DSA`

- If ISL1 has been specified and LOAD=0:
  `RCT=NO;`

- If ISLs have been specified and LOAD=1:
  `RCT=(RCT1nsys,RCT2nsys,RCT3nsys,RCT4nsys);`

* The RSC directive is always generated:
```
-> RSC  nsys
```

- If PID=DSA has not been specified:
  ```
  ADDR=DSA PRTC=DSA200
  ```
- If PID=SID has been specified:
  ```
  ADDR=DSA PRTC=SID
  ```
- If PID=ISO has been specified:
  ```
  ADDR=ISO PRTC=SID PID=PID_SELECT
  ```
- If PID=hex1 has been specified:
  ```
  ADDR=ISO PRTC=SID PID=hex1
  ```
- If NSAP=(hex2 [, hex3] ...) has been specified:
  ```
  NSAP=(hex2 [, hex3] ... )
  ```
- If SCID has been specified:
  ```
  SCID=34:56
  ```
- If ISL1 has been specified :
  ```
  SVR=( ... , TNS)
  RTS=(RTS_nsys)
  ```
- If PT has been specified: for each fsys over a PSI:
  ```
  SVR=( ... , SVR_fsys);
  ```
- If PT has been specified: for each fsys over an ISL:
  ```
  RTS=( ... , RTS_fsys);
  ```

* For ISL1:
```
-> RTS  RTS_nsys NR=(NR41EA14 ...
```

- If ISL2 has been specified :
  ```
  ... NR42EA14 ...
  ```
- If ISL3 has been specified :
  ```
  ... NR33EA13 ...
  ```
- If ISL4 has been specified :
  ```
  ... NR34EA13);
  ```
- If PF={ DPS7 | CNP7 | DN7100 }
  or PF >= { DPS7/GCOS-7/V6 | CNP7/CNS/A2 | DN7100/DN-C/V4.0 }:
  ```
  TP=ISO:2 ;
  ```

* For ISL1:
```
  -> NR   NR41nsys CP=CP_EA14 RPL=RPL1nsys TP=TP_DEFLT;
  -> RPL  RPL1nsys CB=CBL3 RCT=RCT1nsys IADDR=08-00-38-9A-BC-DE;
  -> RCT  RCT1nsys ;
```

- If the TP directive have not already been generated:
```
     -> TP   TP_DEFLT ;
```

* If ISL2 has been specified:
```
  -> NR   NR42psys CP=CP_EA14 RPL=RPL2psys TP=TP_DEFLT;
  -> RPL  RPL2psys CB=CBL3 RCT=RCT2psys IADDR=08-00-38-AB-CD-EF;
  -> RCT  RCT2psys ;
```

* If ISL3 has been specified:
```
  -> NR   NR33psys CP=CP_EA13 RPL=RPL3psys TP=TP_DEFLT;
  -> RPL  RPL3psys CB=CBL2 RCT=RCT3psys IADDR=08-00-38-BC-DE-F0;
  -> RCT  RCT3psys ;
```

* If ISL4 has been specified:
```
  -> NR   NR34psys CP=CP_EA13 RPL=RPL4psys TP=TP_DEFLT;
  -> RPL  RPL4psys CB=CBL2 RCT=RCT4psys IADDR=08-00-38-CD-EF-01;
  -> RCT  RCT4psys ;
```

7.1.1.4 Expansion of Remote System

ISO Addressing

```
SYS rsys TYPE=RMT PF=DPS7 PID=hex1
         [ NSAP=(hex2 [,hex3] ... ) ] ;
```

DSA Addressing

```
SYS rsys TYPE=RMT PF=DPS7 SCID=45:67
         PT=( fsys [, fsys] ...) ] ;
```

Expansion of SYSTEM Directive for Remote System:

* The RSYS directive is always generated:
```
  -> RSYS  rsys TYPE=RMT PF=DPS7 RSC=rsys
```

- If PID=ISO or hex1 has been specified:
```
           ADDR=ISO ;
```

- If PID=DSA or SID has not been specified:
```
           ADDR=DSA ;
```

* The RSC directive is always generated:
```
-> RSC  rsys
```

- If PID=DSA has not been specified:
  `ADDR=DSA PRTC=DSA200`
- If PID=SID has been specified:
  `ADDR=DSA PRTC=SID`
- If PID=ISO has been specified:
  `ADDR=ISO PRTC=SID PID=PID_SELECT`
- If PID=hex1 has been specified:
  `ADDR=ISO PRTC=SID PID=hex1`
- If NSAP=(hex2 [, hex3] ...) has been specified:
  `NSAP=(hex2 [, hex3] ... )`
- If SCID has been specified:
  `SCID=45:67`
- If PT has been specified: for each rsys over a PSI:
  `SVR=( ... , SVR_fsys);`
- If PT has been specified: for each rsys over an ISL:
  `RTS=( ... , RTS_fsys);`
- If PT has been specified: for each rsys over FDDI:
  `SVR=(OCS_EAxx, ...);`

7.1.2 Convention for Naming Layered Objects

The convention for naming layered objects is based on the arguments explicitly specified or assumed by default for parameters in the SYSTEM directive. In the following description, the type of the system to which the SYSTEM directive pertains, is clearly indicated. The object is declared by a directive of the same name, and where it appears as an argument, its corresponding directive is specified.

CB Up to 4 CB objects if the local system is attached over an ISL.

CB=cbl-name: in the LPL directive, comes from the ISL parameter and is:
for the Local System, the third argument:
ISLn=(...dh-hh-hh ,EAhexa2 *[,<u>cbl-name</u>]*
for passthrough or neighbor, the second argument:
ISLn=(...dh-hh-hh *[,<u>cbl-name</u>]*

Defaults:
`CB=CBL MAIN:` for ISL1
`CB=ISL1 :` for ISL2
`CB=ISL2 :` for ISL3
`CB=ISL3 :` for ISL4

CP

One CP object per ISL attachment to the local system.

CP=CP_EAhh: in the NR directive, where *EAhh* comes from the second argument of the ISL parameter for the Local System:
ISLn=(...dh-hh-hh ,*EAhexa2* [,cbl-name].

LCT

One LCT object per PSI or ISL attachment.

LCT=EAhh: in the LPL directive, where *EAhh* comes from the second argument of the relevant ISL parameter:
ISLn=(...dh-hh-hh ,*EAhexa2* [,cbl-name].
LCT=CChh: in the SVR directive, where *CChh* is that specified for the PSI parameter for the Passthrough:
PSI=(*CChexa2 [,CChexa2]*).

LPL

One LPL object per ISL attachment.

LPL=LPL_EAhh: in the CP directive, where *EAhh* comes from the second argument of the ISL parameter of the Local System:
ISLn=(...dh-hh-hh ,*EAhexa2* [,cbl-name].

LSC

LSC=lsys: in the LSYS directive comes from the name given to the Local System.

LSYS

lsys: comes from the name given to the Local System.

LT

Normally the ISO LT and RM LT objects are not expanded by the SYSTEM directive. Whenever if TCPIP=1 is specified in the SYSTEM directive describing the Local System, five LTs per ISL attachment are expanded.
LT in the LCT directive has the value:

LT=RMLTEAhh: for RM LT
LT=ISO1EAhh: for the first ISO LT
LT=ISO2EAhh: for the second ISO LT
LT=TCP1EAhh: for the first TCPIP LT
LT=TCP2EAhh: for the second TCPIP LT

EAhh comes from the second argument of the ISL parameter of the Local System:

ISL*n*=(...dh-hh-hh ,*EAhexa2* [,cbl-name].

NET *lsys:* comes from the name given to the Local System.

NR One NR object per route between the Local System and its neighbor over the ISL.

NR=NRijnsys: in the RTS directive, where
i is *n* of the ISL*n* parameter of the Local System
j is *n* of the ISL*n* parameter of the passthrough or neighbor
nsys is the name given to the passthrough or neighbor.

RCT One RPL object per pass-through or neighbour system attachment over an ISL.

RCT=RCTjnsys: in the RSYS directive, where
j is *n* of the ISL*n* parameter of the passthrough or neighbor
nsys is the name given to the passthrough or neighbor.

RPL One RPL object per passthrough or neighbor attached over the ISL.

RPL=RPLjnsys: in the NR directive, where
j is *n* of the ISL*n* parameter of the passthrough or neighbor
nsys is the name given to the passthrough or neighbor.

RSC *rsys:* comes from the name given to the passthrough, or neighbor or remote system.

RSYS *rsys:* comes from the name given to the passthrough, or neighbor or remote system.

RTS One RTS object per system attached over the ISL.

RTS=RTS_nsys: in the RSC directive, where
nsys is the name given to the passthrough or neighbor.

SVR	One SVR object per passthrough over a PSI.
	SVR=SVR_psys: in the RSC directive, where *psys* is the name given to the passthrough.
	SVR=TNS: in the RSC directive if the ISL1 parameter is specified for the Local System.
TP	One TP object for one paththrough or common to several paththroughs.
	In the NR directive,
	TP=TP_DEFLT: ***Default***
	TP=TP_Dname, where *name* comes from the DEF directive specified with CLASS=TP
	TP=TP_Spsys, where *psys* is the name given to the passthrough if the DEF directive specified with CLASS=TP does not have a name.

7.1.3 Expansion Overriding

When expanding each SYSTEM directive, NETGEN checks if low-level directives appear:

- either between the preceding SYSTEM directive and the directive currently being processed
- or from the beginning of the source for the previous SYSTEM directive.

Two kinds of low-level directives are to be considered:

- those directives which have the same name as the low-level directives which would result from the expansion of the current SYSTEM directive, will affect their expansion
- those which do not correspond to any low-level directives expanded by the SYSTEM directive cannot interfere with this expansion and serve as adding insertions to be discussed later.

If such low-level directives do not exist, the SYSTEM directive is expanded normally.

For overriding to be effective, the following rules must be adhered to:

- All *overriding* low-level directives to be applied to a given system must appear immediately before the SYSTEM directive concerned. Other *overriding* low-level directives may be entered in any order. For the local system, *overriding* low-level directives must come after the NETWORK directive which then becomes mandatory.
- Low-level directives specified *override* those which result from high-level expansion if their names match.
- A low-level directive which does not match one expanded from a given SYSTEM directive will be included if it references objects that are introduced by *overriding* directives of the same SYSTEM directive. Otherwise, the results of including such a non-matching directive may be unpredictable.
- All *overriding* low-level directives must follow the naming conventions used in the expansion of the SYSTEM directive. In general, the *overriding* directive is an extension of the one expanded but with additional parameters and values.
- Default values set by the DEF directive do not apply to low-level directives. If such defaults affect an *overridden* low-level directive, the keyword and value corresponding to those defaults must be explicitly specified in the *overriding* low-level directive if it is to be taken into account.

7.2 Default overriding

Whereas the expansion overriding feature affects the expansion of only one SYSTEM directive, the default overriding feature intends to be remanent for subsequent SYSTEM directives.

For a given SYSTEM directive, expansion overriding always prevails over default overriding which only affects low-level directives actually expanded.

The DEF directive syntax is given in Section 3. Its specific action on the SYSTEM directive is as follows:

TS Class Defaults:

When the class is reset or not set at all, the usual RTS low-level directive resulting from the SYSTEM directive for a passthrough or neighbor, is of the form:

```
-> RTS  RTS_nsys NR=(NRijEAnn [,NRijEAnn ] ... );
```

If either SILENT or FLOW or RETRY defaults have been set, any subsequent SYSTEM directive relating to a passthrough or neighbor will generate its RTS low-level directive as follows:

```
-> RTS  RTS_nsys NR=(NRijEAnn [,NRijEAnn ] ... )
```

- If SILENT has been set at n1:
  `SILENT=n1`
- If FLOW has been set at n2:
  `FLOW=n2`
- If RETRY has been set at n3:
  `RETRY=n3;`

CT Class Defaults:

When the class is reset or not set at all, the usual LCT low-level directive resulting from the SYSTEM directive for a passthrough linked to the local system over a PSI channel is of the form:

```
-> LCT  CC02 TYPE=PSI;
```

If the WATCH default has been set, any subsequent SYSTEM directive relating to a passthrough over a PSI will generate its LCT low-level directives as follows:

```
-> LCT  CC02 TYPE=PSI
```

- If WATCH has been set at n1:
  `WATCH=n1;`

Likewise when the class is reset or not set at all, the usual RCT low-level directive resulting from the SYSTEM directive for a passthrough or neighbor over an ISL is of the form:

```
-> RCT  RCTipsys;
```

If the RMTLOAD default has been set, any subsequent SYSTEM directive relating to a passthrough or neighbor over an ISL will generate its RCT low-level directive:

```
-> RCT  RCTipsys
```

- If RMTLOAD has been set at 0:
  ```
  RMTLOAD=0;
  ```

TP Class Defaults:

When the class has not been set at all, the usual TP low-level directive resulting from the SYSTEM directive and relating to the first encountered passthrough or neighbor attached to the ISL is of the form:

- Before any change of the TP defaults:
  ```
  -> TP TP_DEFLT;
  ```

Without any TP class default setting, this TP low-level directive will be the only one generated and all expanded NR low-level directives will refer to it.
After a reset of the TP class defaults:

- If the DEFAULT directive bears the name tpnm:
  ```
  ->   TP TP_Dtpnm;
  ```

- If the DEFAULT directive has not a name:
  ```
  ->   TP TP_Snsys;
  ```

If either WINDOW or TPDUSIZE defaults have been set, the first encountered SYSTEM directive relating to a passthrough or neighbor over the ISL will generate a new TP low-level directive as follows:

- If the DEFAULT directive bears NAME=tpnm:
  ```
  -> TP  TP_Dtpnm
  ```

- If the DEFAULT directive has not a name:
  ```
  -> TP  TP_Snsys
  ```

- If WINDOW has been set at n1:
  ```
  WINDOW=n1
  ```

- If TPDUSIZE has been set at n2:
  ```
  TPDUSIZE=n2;
  ```

Between two TP low-level directive expansions or between the last TP expansion until the end of the current configuration description, all SYSTEM directive expansions dealing with passthroughs and neighbors will refer to this last expanded TP.

Values set by the DEFAULT directive apply only to the low-level directives directly resulting from the SYSTEM directives. They are completely ignored by other low-level directives inserted among SYSTEM directives. If defaults affect such inserted low-level directives, they must be explicitly specified to be taken into account.

7.3 Adding Insertion

Any low-level directive which is not recognized by any SYSTEM directive, is simply added on and therefore does not interfere with the expansion of SYSTEM directives.

Low-level directives such as LMBX and LT which do not belong to any SYSTEM expansion may be freely inserted in the configuration description.

The low-level directive SR declares after the corresponding SYSTEM is declared.

Any low-level directive which intends to override some SYSTEM expansion but is not placed immediately before the SYSTEM directive concerned, will generate duplicate conflicting objects.

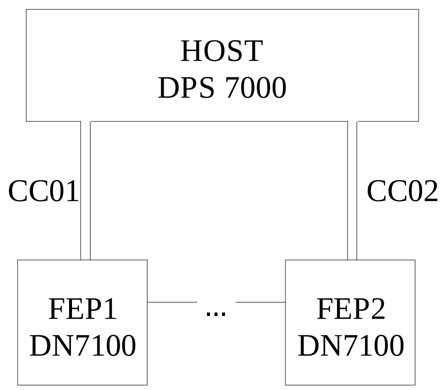
Adding insertions are directives describing:

- queues and RAEH administrative objects which appear in the basic network configuration description after the last SYSTEM directive
- the service of correspondents added in the directory configuration description.

7.4 Examples for Improving the Network Description

7.4.1 Difficulty in Precisely Describing the Network

Consider the following part of a network configuration:



In DSA addressing, such part of network configuration would normally be represented by the following SYSTEM directives:

```
SYS HOST TYPE=LOC SCID=...;
SYS FEP1 TYPE=PT PF=DN7100 SCID=... PSI=CC01 PT=FEP2;
SYS FEP2 TYPE=PT PF=DN7100 SCID=... PSI=CC02 PT=FEP1;
```

Since a SYSTEM directive with the PT parameter must appear after all the systems listed in each PT parameter, the SYSTEM FEP1 directive being first does not comply with this requirement and the compilation result for system FEP1 would have been:

```
SYS FEP1 PF=DN7100 SCID=... PSI=CC01 PT=FEP2;
-> RSYS FEP1 TYPE=PT ADDR=DSA PF=DPS7 SVR=NO RSC=FEP1
-> RSC  FEP1 TYPE=PT SCID=... SVR=(SVR_FEP1,SVR_FEP2);
?  ***  Passthrough system not previously declared: FEP2
```

The same would apply if the SYSTEM FEP2 directive was specified before the SYSTEM FEP1 directive.

This anomaly can be resolved by the following expansion overriding insertion:

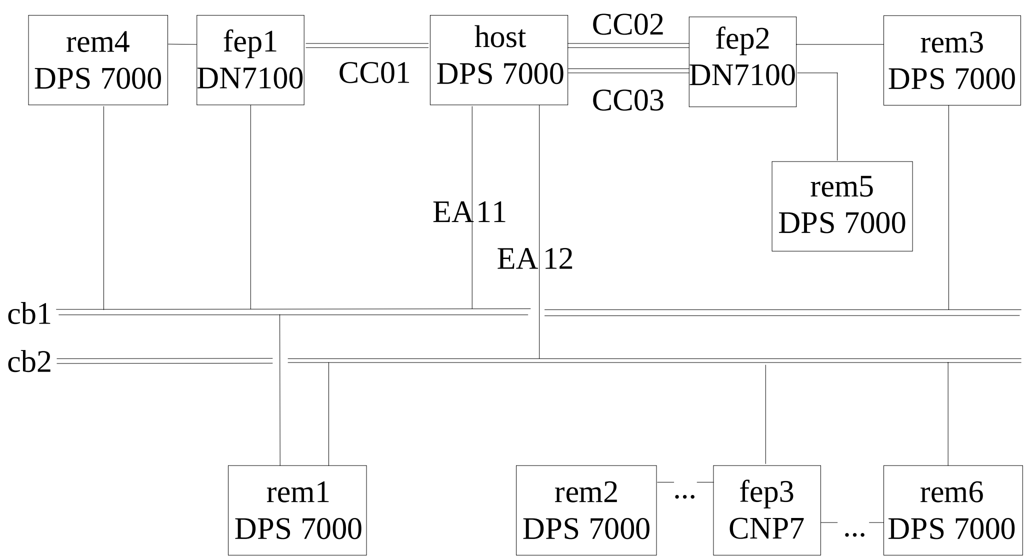
```
SYS HOST PF=LSYS SCID=...;
RSC FEP1 TYPE=PT SCID=... SVR=(SVR_FEP1,SVR_FEP2);
SYS FEP1 PF=DN7100 SCID=... PSI=CC01;
SYS FEP2 PF=DN7100 SCID=... PSI=CC02 PT=FEP1;
```

Here the inserted RSC low-level directive has an expansion overriding effect since it replaces the corresponding RSC low-level directive normally expanded by the SYSTEM FEP1 directive which follows it. So the compilation of system FEP1 becomes error free:

```
RSC FEP1 TYPE=PT SCID=... SVR=(SVR_FEP1,SVR_FEP2);
SYS FEP1 PF=DN7100 SCID=... PSI=CC01;
-> RSYS FEP1 TYPE=PT ADDR=DSA PF=DPS7 SVR=NO RSC=FEP1;
-> RSC  FEP1 is overridden.
```

7.4.2 Improving the General Network Description

7.4.2.1 Network Schematic Description



7.4.2.2 Network Configuration Description

Points worth Noting:

- The specification of two LTs for the local controller EA11 shows how expansion overriding (LCT EA11) can cooperate with adding insertion (LT lt1 and LT lt2) to provide a specific improvement to the network description.
- Each time SYSTEM expansion detects that an expanded low-level directive is to be overridden, OBJLIST reporting:
  - matches the name of the replacing directive with the name of the directive being replaced
  - replaces the parameter values
  - and flags them with *is overridden*.
- The overriding low-level directive RTS named RTS_rem3 shows how expansion overriding prevails over the TS class default overriding.

```
NET  host ;
LSC  host     SCID=45:123 TSAP=abcd;
SVR  TNS      TYPE=TNS EXT=60 DEC=37;
LCT  EA11     TYPE=ISL LT=(lt1,lt2);
LT   lt1      LSAP=ISO TIMER=27;
LT   lt2      LSAP=RM  MAXCCE=72;
LPL  LPL_EA12 LCT=EA12 IADDR=08-00-38-23-45-67 CB=cb2 ST=LOCK;
SYS host  PF=LSYS  SCID=45:123  ISL =(12-34-56,EA11,cb1)
          OBJLIST,              ISL2=(23-45-67,EA12,cb2);
-> NET  host  is overridden
-> LSYS host  ADDR=DSA PF='DPS7/GCOS-7/V10' LSC=host;
-> LSC  host  is overridden
-> SVR  TNS   is overridden
-> CP   CP_EA11  LPL=LPL_EA11;
-> LPL  LPL_EA11 LCT=EA11 IADDR=08-00-38-12-34-56 CB=cb1;
-> LCT  EA11 is overridden
-> CP   CP_EA12  LPL=LPL_EA12;
-> LPL  LPL_EA11 is overridden
-> LCT  EA12     TYPE=ISL;

LCT  CC01     TYPE=PSI,MAXCN=1500;
TP   TP_DEFLT TPDUSZ=512;

SYS fep1  PF=DN7100 SCID=46:122  ISL=(34-56-78,cb1)
          PSI= CC01 OBJLIST;
-> SVR  SVR_fep1 TYPE=FEPS LCT=CC01;
-> LCT  CC01 is overridden
```

```
-> RSYS fep1 ADDR=DSA PF='DN7100/DN-C/V4.1' RSC=fep1
             SVR=SVR_fep1;
-> RSC  fep1 SCID=46:122 ADDR=DSA PRTC=DSA200
             SVR=(TNS,SVR_fep1)   RTS=RTS_fep1;
-> RTS  RTS_fep1 NR=NR11fep1 TP=ISO:2;
-> NR   NR11fep1 CP=CP_EA11 RPL=RPL1fep1 TP=TP_DEFLT;
-> RPL  RPL1fep1 IADDR=08-00-38-34-56-78 CB=cb1 RCT=RCT1fep1;
-> RCT  RCT1fep1 ;
-> TP   TP_DEFLT is overridden

DEF CT    SET WATCH=222;
SYS fep2  PF=DN7100  SCID = 47:121  PSI=(CC02,CC03)
          POOLSZ=(2,5) OBJLIST;
-> SVR  SVR_fep2 TYPE=FEPS LCT=(CC02,CC03) BUFSZ=32,
                 BUFNB=(4096,10240);
-> LCT  CC02     TYPE=PSI,WATCH=222;
-> LCT  CC03     TYPE=PSI,WATCH=222;
-> RSYS fep2     ADDR=DSA PF='DN7100/DN-C/V4.1' RSC=fep2
                 SVR=SVR_fep2;
-> RSC  fep2     SCID=47:121 ADDR=DSA PRTC=DSA200 SVR=SVR_fep2;

NR   NR21fep3 CP=CP_EA12 RPL=RPL1fep3 TP=tp_other;
TP   tp_other TPDUSZ=128;

DEF TS    SET SILENT=111;
SYS fep3  PF=CNP7  SCID=48:120  ISL1=(56-78-9A,cb2) OBJLIST;
-> RSYS fep3     ADDR=DSA PF='CNP7/CNS7/A2' RSC=fep3
                 RCT=(RCT1fep3);
-> RSC  fep3     SCID=48:120 ADDR=DSA PRTC=DSA200 SVR=TNS
                 RTS=RTS_fep3;
-> RTS  RTS_fep3 NR=(NR21fep3) TP=ISO:2 SILENT=111;
-> NR   NR21fep3 is overridden
-> RPL  RPL1fep3 IADDR=08-00-38-56-78-9A CB=cb2
                 RCT=RCT1fep3;
-> RCT  RCT1fep3;

SYS rem1  PF=DPS7  SCID=44:120  ISL =(45-67-89,cb1),
          OBJLIST               ISL2=(78-9A-BC,cb2);
-> RSYS rem1     ADDR=DSA PF='DPS7/GCOS-7/V10' RSC=rem1;
-> RSC  rem1     SCID=44:120 ADDR=DSA PRTC=DSA200 SVR=TNS
                 RTS=RTS_rem1;
-> RTS  RTS_rem1 NR=(NR11rem1,NR22rem1) TP=ISO:2 SILENT=111;
-> NR   NR11rem1 CP=CP_EA11 RPL=RPL1rem1 TP=TP_DEFLT;
-> NR   NR22rem1 CP=CP_EA12 RPL=RPL2rem1 TP=TP_DEFLT;
-> RPL  RPL1rem1 IADDR=08-00-38-45-67-89 CB=cb1
                 RCT=RCT1rem1;
-> RPL  RPL2rem1 IADDR=08-00-38-78-9A-BC CB=cb2
                 RCT=RCT2rem1;
-> RCT  RCT1rem1 ;
-> RCT  RCT2rem1 ;
```

```
RTS  RTS_rem3 NR=NR11rem3 TP=ISO:2 SILENT=456;
NR   NR11rem3 CP=CP_EA12  RPL=RPL1rem3 TP=tp_other;

SYS rem3  PF=DPS7   SCID=42:118 IS =(32-54-76,cb2)
          PT=fep2   OBJLIST;
-> RSYS rem3     ADDR=DSA PF='DPS7/GCOS-7/V10' RSC=rem3;
-> RSC  rem3     SCID=42:118 ADDR=DSA PRTC=DSA200
                 SVR=(SVR_fep2,TNS) RTS=RTS_rem3;
-> RTS  RTS_rem3 is overridden
-> NR   NR11rem3 is overridden
-> RPL  RPL1rem3 IADDR=08-00-38-32-54-76 CB=cb1
        RCT=RCT1rem3;
-> RCT  RCT1rem3 ;

SYS rem4  PF=DPS7   SCID=41:117  ISL1=(79-AB-CD,cb1)
          PT=fep1   OBJLIST;
-> RSYS rem4     ADDR=DSA PF='DPS7/GCOS-7/V10' RSC=rem4;
-> RSC  rem4     SCID=41:117 ADDR=DSA PRTC=DSA200
                 SVR=(TNS,SVR_fep1) RTS=RTS_rem4;
-> RTS  RTS_rem4 NR=NR11rem4 TP=ISO:2 SILENT=111;
-> NR   NR11rem4 CP=CP_EA11 RPL=RPL1rem4 TP=TP_DEFLT;
-> RPL  RPL1rem4 IADDR=08-00-38-79-AB-CD CB=cb1
        RCT=RCT1rem4;
-> RCT  RCT1rem4;

DEF TS    RESET;
SYS rem6  PF=DPS7    SCID=39:115 ISL=(7A-BC-DE,cb2)
          PT=fep3    OBJLIST;
-> RSYS rem6     ADDR=DSA PF='DPS7/GCOS-7/V10' RSC=rem6;
-> RSC  rem6     SCID=41:117 ADDR=DSA PRTC=DSA200
                 SVR=(TNS,SVR_fep3) RTS=RTS_rem6;
-> RTS  RTS_rem6 NR=NR21rem6 TP=ISO:2;
-> NR   NR21rem6 CP=CP_EA12 RPL=RPL1rem6 TP=TP_DEFLT;
-> RPL  RPL1rem6 IADDR=08-00-38-7A-BC-DE CB=cb2
        RCT=RCT1rem6;
-> RCT  RCT1rem6 ;
DIR;
SDOM  PFX=123   NB=(fep2, rem3);
SDOM  PFX=12345 NB=(fep1:1:4);
SDOM  PFX=23    NB=(fep3:2, rem3:3:5, rem1:0);

SYS rem2 PF=DPS7 SCID=43:119  PT=fep3 OBJLIST;
-> RSYS rem2     ADDR=DSA PF='DPS7/GCOS-7/V10' RSC=rem2;
-> RSC  rem2     SCID=43:119  SVR=TNS  RTS=RTS_fep3;

SYS rem5 PF=DPS7 SCID=40:116 PT=(fep2) OBJLIST;
-> RSYS rem5     ADDR=DSA PF='DPS7/GCOS-7/V10' RSC=rem5;
-> RSC  rem5     SCID=41:117 SVR=SVR_fep2;

EDIR;
```

For a system with OCS (only for DPS 7000):

```
SYS host PF=LSYS SCID=81:015 NCC1=(08-00-38-53-51-0F,EA31)
    OBJLIST;
-> SYSTEM NAME=host PF=LSYS SCID=81:015 OBJLIST=1
          NCC1=(08-00-38-53-51-0F EA31);
-> NET    NAME=host;
-> LSYS   NAME=host ADDR=DSA PF='DPS7' LSC=host;
-> LSC    NAME=host SCID=81:015 ADDR=DSA PRTC=DSA200;
-> LCT    NAME=EA31 TYPE=NCC IADDR=08-00-38-53-51-0F;
-> SVR    NAME=OCS_EA31 TYPE=OCS LCT= EA31 BUFSZ=128
          BUFNB=(7680 16384);

SYS rem5  PF=DPS7  SCID=40:116 PT=(EA31) OBJLIST;
-> RSYS rem5     ADDR=DSA PF='DPS7/GCOS-7/V10' RSC=rem5;
-> RSC  rem5     SCID=41:117 SVR=OCS_EA31;
```

For a system with OCS (for DPS 7000/XTA):

```
SYS host PF=LSYS SCID=81:015 NCC1=(00-90-27-C3-2F-C6,EA31)
    OBJLIST;
-> SYSTEM NAME=host PF=LSYS SCID=81:015 OBJLIST=1
          NCC1=(00-90-27-C3-2F-C6 EA31);
-> NET    NAME=host;
-> LSYS   NAME=host ADDR=DSA PF='DPS7' LSC=host;
-> LSC    NAME=host SCID=81:015 ADDR=DSA PRTC=DSA200;
-> LCT    NAME=EA31 TYPE=NCC IADDR=00-90-27-C3-2F-C6;
-> SVR    NAME=OCS_EA31 TYPE=OCS LCT= EA31 BUFSZ=128
          BUFNB=(7680 16384);

SYS rem5  PF=DPS7  SCID=40:116 PT=(EA31) OBJLIST;
-> RSYS rem5     ADDR=DSA PF='DPS7/GCOS-7/V10' RSC=rem5;
-> RSC  rem5     SCID=41:117 SVR=OCS_EA31;
```

```
NET BC05 ;

COMM '-----------------------------------------------------------------';
COMM ' Local system DPS 7000 XTA, for MainWay 2600LE Administration  ';
COMM '-----------------------------------------------------------------';

   SYS BC05 PF=LSYS SCID=65:005
            ISL =(00-90-27-FC-16-F3,EAD1,P0)
            MADDR=(00-01-00-00-00-00, 7F-FF-FF-FF-FF-FF)
            NCC1=(00-03-47-4C-58-AB,EA11)
            NCC2=(00-90-27-C3-2F-C6,EA21)
            NCC3=(00-90-27-C2-C7-44,EA31)
            NCC4=(00-90-27-FC-16-F2,EA41)
            OBJPFX=0 OBJLIST ;

-> SYSTEM NAME=BC05 PF=LSYS SCID=65:005 ISL1=(00-90-27-FC-16-F3 EAD1 P0)
   OBJLIST=1   OBJPFX=0
   MADDR=(00-01-00-00-00-00 7F-FF-FF-FF-FF-FF)
   NCC1=(00-03-47-4C-58-AB EA11)
   NCC2=(00-90-27-C3-2F-C6 EA21)
   NCC3=(00-90-27-C2-C7-44 EA31)
   NCC4=(00-90-27-FC-16-F2 EA41);
-> NET    BC05 is overriden.
-> LSYS   NAME=BC05 ADDR=DSA PF='DPS7' LSC=BC05;
-> LSC    NAME=BC05 SCID=65:005 ADDR=DSA PRTC=DSA200;
-> SVR    NAME=TNS  BUFSZ=128 BUFNB=(512 2048);
-> CP     NAME=EAD1 LPL=EAD1;
-> LPL    NAME=EAD1 LCT=EAD1 IADDR=00-90-27-FC-16-F3 CB=P0
          MADDR=(00-01-00-00-00-00 7F-FF-FF-FF-FF-FF);
-> LCT    NAME=EAD1;
-> LCT    NAME=EA11 TYPE=NCC IADDR=00-03-47-4C-58-AB;
-> SVR    NAME=EA11 TYPE=OCS LCT=EA11 BUFSZ=128 BUFNB=(7680 16384);

-> LCT    NAME=EA21 TYPE=NCC IADDR=00-90-27-C3-2F-C6;
-> SVR    NAME=EA21 TYPE=OCS LCT=EA21 BUFSZ=128 BUFNB=(7680 16384);


-> LCT    NAME=EA31 TYPE=NCC IADDR=00-90-27-C2-C7-44;
-> SVR    NAME=EA31 TYPE=OCS LCT=EA31 BUFSZ=128 BUFNB=(7680 16384);

-> LCT    NAME=EA41 TYPE=NCC IADDR=00-90-27-FC-16-F2;
-> SVR    NAME=EA41 TYPE=OCS LCT=EA41 BUFSZ=128 BUFNB=(7680 16384);
```

```
COMM '-----------------------------------------------------------';
COMM ' MainWay 2600LE Administrated by DPS 7000 XTA             ';
COMM '-----------------------------------------------------------';

SYS BC86 PF=MW2600LE TYPE=PT SCID=65:134 ISL=(08-00-38-11-00-45,P0)
    PT=(EA41) LOAD OBJLIST;

-> SYSTEM NAME=BC86 TYPE=PT PF=MW2600LE SCID=65:134 PT=(EA41)
-> ISL1=(08-00-38-11-00-45 P0) LOAD=1 OBJLIST=1;
-> RSYS   NAME=BC86 TYPE=PT ADDR=DSA PF='MW2600LE' RSC=BC86 RCT=(BC86);
-> RSC    NAME=BC86 TYPE=PT SCID=65:134 ADDR=DSA PRTC=DSA200
          SVR=(EA41 TNS) RTS=(BC86);
-> NR     NAME=BC86 CP=EAD1 RPL=BC86 TP=TP_DEFLT;
-> RPL    NAME=BC86 IADDR=08-00-38-11-00-45 CB=P0 RCT=BC86;
-> RCT    NAME=BC86 TYPE=MWLE;
-> RTS    NAME=BC86 NR=(BC86);



COMM '------------------------------------------------------------';
COMM ' TNS server declaration, for MainWay 2600LE Administration ';
COMM '------------------------------------------------------------';
SVR   TNS  TYPE=TNS;
LCT   EAD1;
LPL   EAD1  LCT=EAD1  IADDR=00-90-27-FC-16-F3  CB=P0;
CP    P0  LPL=EAD1;
TP    PRTC1;

COMM '---------------------------';
COMM '  OCS server declaration    ';
COMM '---------------------------';
SVR   OCS4  TYPE=OCS  LCT=EA41;
LCT   EA41  TYPE=NCC  IADDR=00-90-27-FC-16-F2;



COMM '------------------------------------------';
COMM '       MainWay 2600LE BC86 declaration    ';
COMM '------------------------------------------';
RSYS BC86  RSC=BC86  PROFILE='MW2600LE/DNS-E/V3'  RCT=BC86;
RSC  BC86  SCID=065:134  SERVER=(TNS,OCS4)  RTS=IC86;
RTS  IC86  NR=IC86;
NR   IC86  CP=P0  TP=PRTC1  RPL=IC86;
RPL  IC86  IADDR='08-00-38-11-00-45'  RCT=BC86  CB=P0;
RCT  BC86;
```

8. OCS Front End Configuration

8.1 Overview of OCS Front End Configuration

Configuration is the process of setting up network parameters to enable the local DPS 7000 host to communicate with the network through the OCS Front End.

This process is automatically initiated after the OCS server has been started and the corresponding front end is booted by means of the STSVR operator command (see the STSVR command in the *Network User Guide*).

To perform the above function, a set of DPS 7000 system subfiles is defined and a set of commands is provided. They are seen in details for organization and syntax respectively, showing how they act during the configuration process regardless of the way that process is launched.

Attached to each OCS Front End, a set of subfiles contains command lines allowing the configuration of both IPS and OSI/DIWS stacks.

These subfiles are in the SYS.DSACONF library.

The IPS and OSI/DIWS stacks are configured separately. Each of them has its own configuration subfiles and commands (except when RFC1006 is used).

Only IPS administration commands can be used to modify the IPS stack configuration when the OCS server is connected to the network.

The OSI/DIWS configuration can be modified dynamically using administrative commands either to create or select an addressing or routing element, or by using the incremental configuration mechanism (see paragraph 8.1.4).

To facilitate the construction of the OCS Front End configuration subfiles, the following commands are available (and described in Section 8):

BUILD_FCP_CONFIG to build a OCS Front End configuration subfile,

VERIFY_FCP_CONFIG to verify a OCS Front End configuration subfile.

OCS Front End contains two communications stacks:

- OSI stack (containing OSI layers 1 to 4)
- IPS stack (containing TCP, UDP, ICMP and IP layers)

The OSI stack allows the use of an ISO transport (layer 4) over either a Full Internet (layer 3) or Null Internet (layer 3). The OSI stack is also referred to as the OSI/DIWS stack.

The RFC1006 layer allows OSI/DIWS sessions to be run over the IPS stack of OCS Front End.

8.1.1 Configuration Subfiles

The GO order in the scenario subfile gives the name(s) of the configuration subfile(s). If both OSI/DIWS and IPS are being configured, each has a separate configuration subfile.

In the example of Figure 8-1, the names of the configuration subfiles are EA92_CONF_ISO (for OSI/DIWS) and EA92_CONF_IPS (for IPS). Note how the "CONF" prefix of the name (given in the GO order) is removed to derive the subfile name.

You must place the configuration commands in the appropriate subfile(s).

In the case of IPS, the ETC_HOSTS, ETC_NETWORKS, and ARPCO_arpfile subfiles are optional. However, if you use a symbolic name for a network ("abc" in the example of Figure 8-1), then the ETC_NETWORKS subfile must exist and contain a definition for "abc". Similarly, if you use a symbolic name for a host, the ETC_HOSTS subfile must exist (and contain "xyz" in the example of Figure 8-1).

Example of Configuration Subfiles

Figure 8-1 below gives an overview of the OCS Front End configuration subfiles.

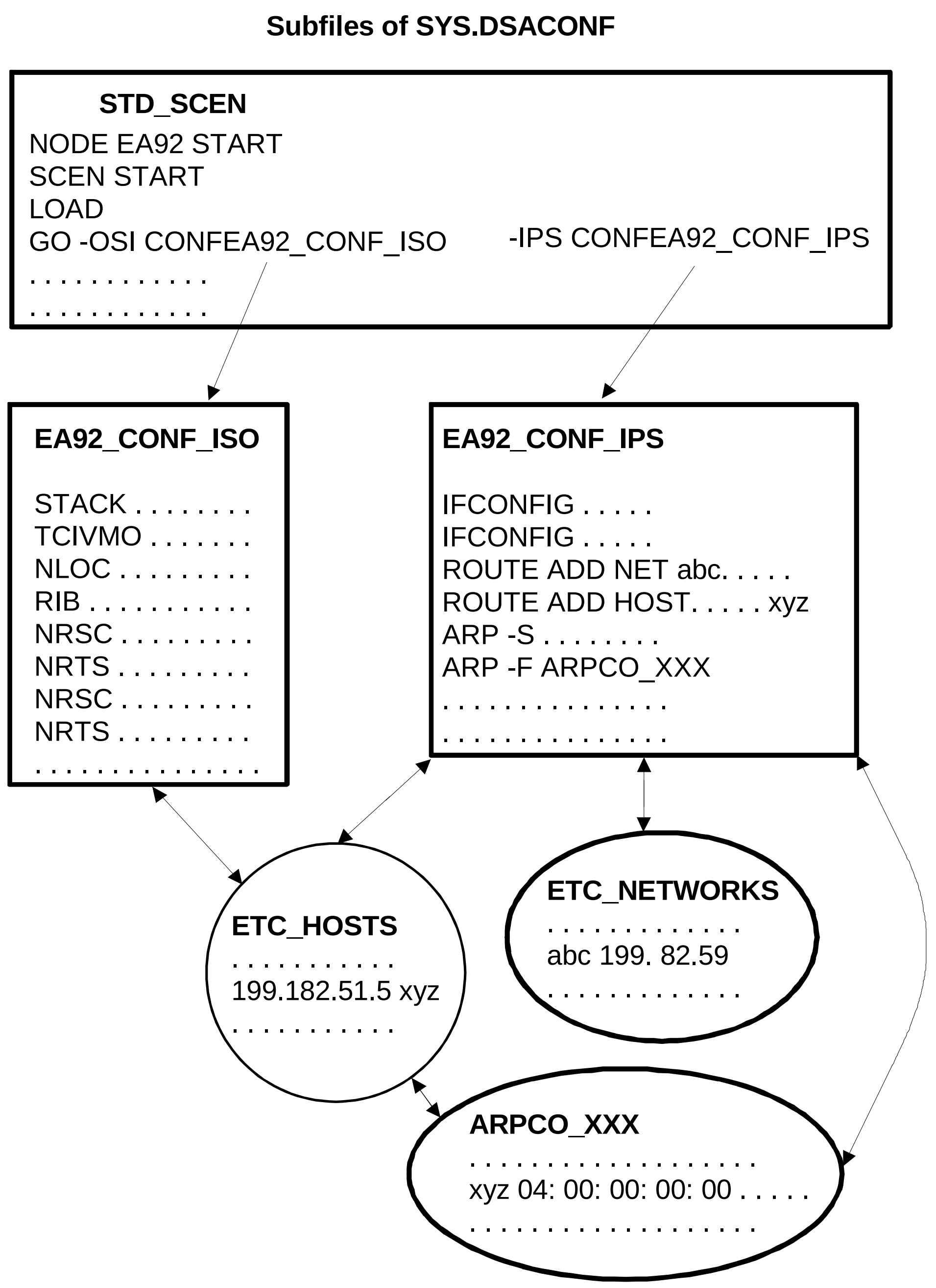


Figure 8-1. Overview of Configuration Subfiles

Explanation of Figure 8-1

All the subfiles are in the library SYS.DSACONF.

The GO order in the STD_SCEN (standard scenario) gives the names of the configuration subfiles. In this example, there are two such subfiles: EA92_CONF_ISO and EA92_CONF_IPS.

EA92_CONF_ISO is for the ISO/DIWS configuration. It contains directives such as STACK, TCIVMO, NLOC, RIB, NRSC, NRTS, NSDOM, LTS_INET, INET_RT, and FILE. To avoid overloading Figure 8-1 with too much detail, the parameters are not shown (they are represented by dots). The directives concerned are described in paragraph 8.3 (later in this chapter).

EA92_CONF_IPS is for the IPS configuration. It contains directives such as IFCONFIG, ROUTE, and ARP. To avoid overloading Figure 8-1 with too much detail, only some of the parameters are shown (the others are represented by dots). The directives concerned are described in paragraph 8.4 (later in this chapter).

Note that the symbolic name "abc" is used in the "ROUTE ADD NET" directive of EA92_CONF_IPS. Therefore "abc" must be present (defined) in the ETC_NETWORKS subfile. Similarly, the symbolic name "xyz" is used in the "ROUTE ADD HOST" directive of EA92_CONF_IPS. Therefore "xyz" must be present (defined) in the ETC_HOSTS subfile. Since the "ARP -f ARPCO_xxx" directive is used in EA92_CONF_IPS, the subfile ARPCO_xxx must also be present.

8.1.2 Configuration Process

Starting

Configuration starts when the OCS server is started (STSVR command or automatically).

The Configuration process is as follows:

1. The Configuration Report subfile is opened (for writing).
2. The Configuration subfiles are opened (for reading). Associated subfiles (if any) are also opened.
3. The configuration commands are executed one at a time.
4. Each command line is checked (semantics and syntax).
5. The GCOS 7 local driver tables are created and loaded (for example, NRTS table, NRSC table, NSDOM table, IVMO table, NSAP table, INETRT table).
6. Some configuration commands are also sent to the stacks in the FCP7. Some commands (NRSC, NRTS, NSDOM, INET-RT) are local to GCOS 7.

Syntax Checking

For each command:

- It is recorded in the report subfile (located in the GCOS 7 library SYS.DSACONF).
- It is checked for validity.
- A record is built according to command arguments and parameters.
- This record is sent from the DPS 7000 to the OCS Front End concerned, and processing waits for a response to come back.
- The incoming response is analyzed. If an error is notified, it is recorded in the report subfile. An error can be a warning or a fatal error.
- Processing continues until all commands and their corresponding responses are processed unless a fatal error is detected. A single fatal error will stop the processing. Warnings are logged and do not affect the processing.
- If any errors (warning or fatal) are found, an error message is displayed at the operator console.

8.1.3 Configuration Report

The OCS Configuration Report is placed in a subfile of the GCOS 7 library named SYS.DSACONF. The name of this subfile is:

EAnn_CONF_RPRT

The two digits nn string identifies the OCS Front End that the subfile concerns. There is one such subfile for each OCS Front End. The subfile is created at the start of the configuration process.

For a given OCS Front End, the subfile contains information about both the IPS and OSI/DIWS configurations (there are not separate subfiles for the IPS and OSI/DIWS configuration reports). All configuration commands and errors detected during the configuration process (if any) are recorded in the subfile.

This subfile will be salvaged by DP_SAVE, with OCS and RON keywords, in case of a major failure.

8.1.4 Dynamic Configuration of the OSI/DIWS Stack

Dynamic configuration is the process of configuring the OSI/DIWS stack when the OCS server is running. Dynamic configuration is performed by using administrative commands, or by using the incremental configuration mechanism.

Dynamic configuration is possible only if an OSI/DIWS stack has already been configured.

Dynamic configuration via administrative commands allows you to:

- create TCIVMO, NSAP, or RIB entries,
- delete NSAP or RIB entries,
- modify TCIVMO objects,
- create or modify the link between a TCIVMO and an RTS or NSAP.

The incremental configuration function allows you to add new remote systems and new route descriptions to the current OCS Front End configuration (see section 8.4).

A whole OSI/DIWS stack cannot be configured dynamically. In this case, a OCS server restart and scenario subfile modification are necessary.

For the operator commands which apply to the OSI/DIWS stack, see the *Network User Guide.*

8.1.5 Dynamic Configuration of IPS Stack

Dynamic configuration is the process of configuring the IPS stack when the OCS server is running. Dynamic configuration is performed by using administrative commands.

Dynamic configuration is possible only if an IPS stack was already configured at STSVR time.

The IPS administrative command set allows to dynamically modify an already configured IPS Stack.

Like for the OSI stack, the configuration of a running IPS stack can be altered by the creation or deletion of ARP or Route tables and modification of interface parameters (IPaddress, flags, ...).

For the operator commands which apply to the IPS stack, see the *Network User Guide*.

8.2 OSI/DIWS Configuration SUBFile

8.2.1 Naming the Configuration Subfile

There is one OSI/DIWS configuration subfile for each OCS Front End. The OSI/DIWS configuration subfile is located in the GCOS 7 library named SYS.DSACONF. The subfile name is given in GO order of scenario subfile.

For details on the scenario subfile, see the *Network User Guide*.

8.2.2 Naming the Incremental Configuration Subfile

This subfile is located in the SYS.DSACONF library and its name is EAxx_ISOCONF_INCR (where EAxx equals the current OCS Front End name).

8.2.3 Structure of the Configuration Subfile

This subfile contains a list of OSI/DIWS configuration command lines. Command lines are character strings of printable characters (upper and/or lower case can be used). Command lines are separated from each other by means of the new line character.

The subfile may contains blank lines or comments lines. A comments line must begin with the "#" character.

Command name and arguments or parameters are separated by any number of blank characters.

The total length of a command line should not be greater than 255 characters including the new line character.

8.2.4 Contents of the Configuration Subfile

The OSI/DIWS Configuration Subfile is composed of the following records:

- STACK
- TCIVMO
- RIB
- NLOC
- NRSC
- NSDOM
- NRTS
- FILE
- LTS_INET
- INET_RT

Depending on the addressing mode, the NSDOM and NRSC directives allow you to define an access path to the remote systems using the ISO Transport with Null Internet as follows:

NSDOM	ISO addressing
NRSC	DSA addressing

Each directive defines a list of RTSs (Remote Transport Stations) on the access path to be used. The list is scanned until an entry enables a transport connection to be established. The transport connection request is rejected if the end of the list is reached. At OCS Front End transport level, quota and QOS (Quality of Service) are not supported.

8.2.5 Maintaining the Configuration Subfile and the Incremental Configuration Subfile

The Configuration Subfile and the Incremental Configuration Subfile can be created using any GCOS 7 editor. All can be displayed and/or modified the same way.

To more easily create the OCS Front End Configuration Subfile and the Incremental Configuration Subfile, two GCL commands are available in the IOF domain. The GCL command BUILD_FCP_CONFIG builds the OCS Front End Configuration Subfiles. The GCL command VERIFY_FCP_CONFIG performs a first level of command syntax verification of the OCS Front End Configuration Subfiles. See "OCS Front End Configuration Tools" later in this chapter.

8.3 OSI/DIWS CONFIGURATION DIRECTIVES

8.3.1 STACK Directive

Applies only to the Configuration Subfile. Not allowed in an Incremental Configuration Subfile.

Purpose

This record is optional, it defines the parameters of the OSI/DIWS stack of the OCS Front End. This record, if present, is unique for one OCS Front End.

If there is no STACK directive in the Configuration Subfile, a STACK directive is automatically generated at the end of configuration. In this automatic STACK directive, the following values apply: TSNAME = <LCT name>, MAXTC = MAXCN value, and ESIS is enabled. If a STACK directive is given explicitly, MAXTC = MAXCN value (see the MAXCN parameter in the LCT directive in the NG file).

Syntax

```
STACK  [TSNAME = char-8]

       [I_NOERRFLAG]

       [I_CHECKSUM]

       [I_NOSEGFLAG]

       [ESIS = {0 | 1}]

       [CONFTIME = dec-3]

       [HOLDTIME = dec-3]
```

Parameter	**Description**
TSNAME	a string of up to 8 bytes giving the name of the TS (DSAC object: Transport Station). Default name is the OCS Front End name given by the NETGEN directive SVR.
	The following parameters concern ISO only.
I_NOERRFLAG	disables the error report flag (default is to set the flag).
I_CHECKSUM	requests use of checksum (default is no use of checksum).
I_NOSEGFLAG	disables segmentation (default is use of segmentation).
ESIS	validates the ES-IS routing protocol (default is: the routing protocol is enabled).
CONFTIME	value of the configuration timer (default value = 180 sec.). In the ES-IS protocol, this determines the interval between ISH (Intermediate System Hello) and ESH (End System Hello) frames.
HOLDTIME	value of the holding timer (default value = 2). HOLDTIME is expressed in number of CONFTIME value. In the ES-IS protocol, this determines the length of validity of information contained in ISH (Intermediate System Hello) and ESH (End System Hello) frames (2 <= HOLDTIME <= 63).

8.3.2 TCIVMO Directive

Applies both to the Configuration Subfile and the Incremental Configuration Subfile.

Purpose

A TCIVMO (Initial Value Management Object for Transport Connection) is a set of initialization parameters which will be used by the Transport layer at connection request time.

This record is optional. A default IVMO (IVMO_DF) is automatically created at the beginning of ISO and/or DIWS configuration.

Syntax

```
TCIVMO  NAME = char-8

        [MAXRCRDT = dec-3]

        [MAXSCRDT = dec-3]

        [MAXRETRY = dec-1]

        [T1TIME = dec-6]

        [WTIME = dec-6]

        [ITIME =dec-6]

        [ACKTIME = dec-2]

        [TPUSIZE = dec-4]

        [EXTMODE]

        [T_CHECKSUM]
```

Parameter	Description
NAME	up to 8 characters giving the name of the TCIVMO entity used in the NRTS and RIB directive. Note that the name IVMO_DF is reserved for the default TCIVMO directive.
MAXRCRDT	maximum receive credit (default value = 7). If no extended mode, the value must be from 1 up to 15. If extended mode, the value must be from 1 up to 127.
MAXSCRDT	acknowledgement window size or maximum send credit (default value = 7). If no extended mode, the value must be from 1 up to 15. If extended mode, the value must be from 1 up to 127.
MAXRETRY	maximum number of re-transmissions of a PDU which requires acknowledgement (value from 1 up to 9, default value = 3).
T1TIME	acknowledgement time-out before re-transmission expressed in milliseconds. The possible value is from 1000 to 120,000 (but is rounded down internally to a multiple of 1000 milliseconds). The default value is 10,000.
WTIME	window time expressed in milliseconds. The possible value is from 10,000 to 900,000 (but is rounded down internally to a multiple of 10,000 milliseconds). The default value is 60,000.
ITIME	inactivity time expressed in milliseconds. The possible value is from 30,000 to 1,800,000 (but is rounded down internally to a multiple of 10,000 milliseconds). The default value is 180,000.
ACKTIME	acknowledgement retention time expressed in milliseconds. The possible value is from 500 to 60,000 (but is rounded down internally to a multiple of 500 milliseconds). The default value is 5000.
TPUSIZE	maximum TPDU length, expressed in bytes (value = 128 | 256 | 512| 1024 | 2048 | 4096 | 8192, default value = 1024). The following parameters concern ISO only.
T_CHECKSUM	use of checksum (default is no use of checksum).
EXTMODE	extended numbering (default is normal numbering).

8.3.3 RIB Directive

Applies both to the Configuration Subfile and the Incremental Configuration Subfile.

Purpose

This record defines a static entry in the RIB (Route Information Base). This record is useful in the OSI stack configuration, default implies the use of ES-IS protocol.

Syntax

```
RIB  IDENT = char-8

     RNSAP = hexa-40

     [MASK = hexa-40]

     {SNPA = hexa-12 | SNPAM = hexa-40 | NET = hexa-40}

     IVMO = char-8
```

SNPA/SNPAM/NET are mutually exclusive parameters, but one of them is mandatory.

Parameter Description

IDENT	up to 8 characters string, giving the name of the OSI remote entity for management purposes.
RNSAP	remote NSAP address, as a string of hexadecimal digits (up to 40 digits) giving the NSAP address to be reached.
MASK	NSAP_MASK, string of bits, representing the mask, left justified, for an equivalent class (optional, up to 40 digits).
SNPA	string of 12 hexadecimal digits giving the MAC address of the next hop. As an alternative to giving the address explicitly via SNPA, you can give it implicitly via SNPAM or NET.

SNPAM	string of hexadecimal digits defining a mask. This mask is made up of "F" hexadecimal digits on the left, followed by hexadecimal 0s on the right. When the SNPAM mask is applied to the NSAP, it gives the MAC address of the remote system (Full Internet).
NET	string of hexadecimal digits (up to 40 digits) giving the Network Entity Title of the next hop.
IVMO	up to 8 characters, giving the name of the TCIVMO related to this NSAP which will be used at the connection request. If no TCIVMO directive has been configured, only the value IVMO_DF is allowed.

8.3.4 NLOC Directive

Applies both to the Configuration Subfile and the Incremental Configuration Subfile.

Purpose

This record defines a local NSAP address. At least one is mandatory in a OSI configuration. Several can be specified.

Syntax

```
NLOC  NAME = char-8

      LNSAP = hexa-40
```

Parameter Description

NAME	up to 8 character string, giving the name of the entity for management purposes.
LNSAP	string of hexadecimal digits (up to 40 digits) giving the local NSAP address.

8.3.5 NRSC Directive

Applies both to the Configuration Subfile and the Incremental Configuration Subfile.

Purpose

This record describes the ways to access to a Remote Session Control (at least one is mandatory in a DIWS configuration).

Syntax

```
NRSC  NAME = char-4

      NRTS = char-8 [, char-8 ...]
```

Parameter Description

NAME	Identifies the name of the Remote Session Control. It must be the same as the one contained in the NETGEN subfile.
NRTS	A list of Transport Stations that can be used to access to the Remote Session. Each NRTS is identified by its name and it is defined by a NRTS directive. The list can contain up to 20 transport station names. If there are several stations in the list, they are tried in list order (left to right) and the first one that works is used. The others are used as backup.

8.3.6 NSDOM Directive

Applies both to the Configuration Subfile and the Incremental Configuration Subfile.

Purpose

Defines an OSI addressing subdomain in the network. This subdomain is a set of OSI network addresses which have in common some specified upper digits (mapping NSAP prefix) and which follow the same routing rules.

Syntax

```
NSDOM  NAME = char-8

       NSAP_PRF = hexa-40

       NRTS = char-8 [, char-8 ...]
```

Parameter Description

NAME	Identifies the name of the subdomain.
NSAP_PRF	String of up to 40 hexadecimal digits, giving the prefix of OSI Network Service Access Point addresses. It specifies the leftmost part of the NSAP addresses which is typical of the current subdomain.
NRTS	A list of Transport Stations that can be used to access to the subdomain. Each NRTS is identified by its name and it is defined by a NRTS directive. A NRTS list can contain up to 20 names. If there are several stations in the list, they are tried in list order (left to right) and the first one that works is used.

8.3.7 NRTS Directive

Applies both to the Configuration Subfile and the Incremental Configuration Subfile.

Purpose

This record describes a Remote Transport Station located on the LAN in a neighbor or a passthrough system.

Syntax

```
NRTS  NAME = char-8

      TYPE = {DIWS | DSA57 | INET}

      [IVMO = char-8]

      [MACADR = hexa-12]

      [NRSC = char-4 [RSC_TYPE = {PT | NB}]]

      [IP_ADD = {@ip | name}]

      [PORT = {port# | 0102}]

      [GW_ADD = {@ip | name}, METRIC = dec-2]
```

Parameter Description

NAME Identifies the name of a Remote Transport Station.

TYPE Identifies the type of the transport protocol which must be used to access to the Remote Transport Station. The 3 possible values (DIWS, DSA57 and INET) are defined thus:

DIWS means DIWS protocol using ISO/DSA addressing.
It means ISO transport with Null Internet.
It is equivalent to the value ISO:1 in the TPROTOCOL parameter of the RTS directive as applied to TNS.

DSA57 means a specific transport protocol extension to access a remote system. This protocol extension allows you to send a remote ISO address (TSEL + NSAP) over a DIWS transport connection on the OCS Front End to access ISO layers in a pass-thru system (CNP7 or DNS).

	DSA57 is equivalent to the value ISO:2 in the TPROTOCOL parameter of the RTS directive as applied to TNS. **INET** means an RFC1006 transport protocol is used to access a remote system.
IVMO	Identifies the IVMO associated to the route. If no corresponding TCIVMO command has been configured in the OCS Front End configuration, then only the value IVMO_DF is allowed (used only if type = DIWS or DSAS7).
MACADR	Defines the MAC address of the NRTS (used only if type = DIWS or DSAS7).
NRSC	Identifies the RSC name of the remote system, used for the DSA-57 extension.
RSC_TYPE	Type of the remote system: PT: Pass-through system (default value), system located on the LAN which can be used as an intermediate node (for instance MainWay 2000). NB: Neighbor system, system located on the LAN but which cannot used as an intermediate node (for instance DPS 7000).
IP_ADD	IP address (or equivalent name in the ETC_HOSTS subfile) of the RFC1006 node used to access the Remote Transport Station. Used only if TYPE = INET.
PORT	A decimal value giving the TCP port of RFC1006. The default is 102. 1 <= PORT <= 65535. Used only if TYPE = INET.
GW_ADD	Identification of the first gateway required to reach the RFC1006 node when it is not on the same network as the local system. GW_ADD can be a name (which must be declared in the ETC_HOSTS member of the SYS.DSACONF library) or an IP address (expressed in decimal dot notation). Used only if TYPE = INET.
METRIC	Identifies the maximum number of hops required to reach the RFC1006 node. If present, METRIC must be >= 1. Used only if TYPE = INET.

NOTES:

1. IP_ADD can be the host address if the host has the RFC1006 capability.
2. A special NSAP format (called RFC1277 format) has been defined for using OSI applications running on TCP/IP networks. For a description of the standard NSAP address format, see the manual *Network Overview and Concepts*. The RFC1277 format NSAP address is 15 bytes long and is as follows:

IDP	DSP			
5400728722	03	Internet Add	Port #	0

   For example:

   Internet Address: 199.182.51.131

   Port Number: 102

   NSAP Coding: 540072872203199182051131001020

3. An INET_RT directive is not necessary in order to declare a remote system running RFC1006 which has an NSAP address in RFC1277 format.

8.3.8 LTS_INET Directive

Applies only to the Configuration Subfile. Not allowed in an Incremental Configuration Subfile.

Purpose

This record is optional. It defines the parameters of the RFC1006 transport of the OCS Front End. This record, if present, is unique for one OCS Front End.

It is not allowed in a secondary Configuration Subfile (defined via the FILE directive).

If there is one LTS_INET directive in the Configuration Subfile, the MAXTC value given explicitly will change the MAXTC value of the STACK directive. In such a case, the MAXTC value of the STACK directive will be equivalent to MAXCN (NETGEN value) minus MAXTC (of LTS_INET). For the NETGEN value, see the MAXCN parameter in the LCT directive in the NG file.

Syntax

```
LTS_INET  NAME = char-8

      [IP_ADD = {@ip | name}]

      [PORT = {port# | 0102}]

      MAXTC = dec-4

      [TPUSIZE = {dec-5 | 16384}]

      [GW_ADD = {@ip | name}]
```

Parameter Description

NAME	Gives the name (up to 8 bytes) of the Internet Transport Station (DSAC object: Transport Station).
IP_ADD	IP address (or equivalent name in the ETC_HOSTS subfile) of the local system. If omitted, the IP address will be given in the IPS Configuration Subfile (which is mandatory in this case to configure the RFC1006 transport).
PORT	A decimal value giving the TCP port of RFC1006. The default is 102. 1 <= PORT <= 65535.

MAXTC	Gives the maximum number of RFC1006 transport connections. The value must be <= MAXCN (of NETGEN) of the LCT directive attached to the OCS Front End.
TPUSIZE	Defines the maximum TPDU length in bytes. The TPDU length value is a multiple of 128 (except the maximum value of 65531). The default value is 16384.
GW_ADD	IP address (or equivalent name in the ETC_HOSTS subfile) of the gateway used to allow routing of TCP. If omitted, ROUTE entries should be given in the IPS Configuration Subfile.

8.3.9 INET_RT Directive

Applies both to the Configuration Subfile and the Incremental Configuration Subfile.

Purpose

This record is optional.

It defines:

- The route to reach a remote OSI system through an RFC1006 network.
- The node which offers OSI/DIWS to TCP/IP switching (an RFC1006 node).
- The TCP/IP gateway required to access the RFC1006 node (when not on the same network).

Several routes can be defined to access the same remote NSAP.

Syntax

```
INET_RT
      RNSAP = hexa-40
      IP_ADD = {@ip | name}
      [PORT = {port# | 0102}]
      [GW_ADD = {@ip | name}, METRIC = dec-2]
```

Parameter Description

RNSAP	Remote NSAP address (up to 40 hexadecimal digits) giving the NSAP address to be reached.
IP_ADD	IP address (or equivalent name in the ETC_HOSTS subfile) of the RFC1006 node used to access the remote NSAP.
PORT	A decimal value giving the TCP port of RFC1006. The default is 102. 1 <= PORT <= 65535).
GW_ADD	Identification of the first gateway required to reach the RFC1006 node when it is not on the same network as the local system. GW_ADD can be a name (which must be declared in the ETC_HOSTS member of the SYS.DSACONF library) or an IP address (expressed in decimal dot notation).
METRIC	Identifies the maximum number of hops required to reach the RFC1006 node. If present, METRIC must be >= 1.

8.3.10 Automatically Induced Directives (IPS Configuration)

It is not necessary to explicitly configure the IPS stack of the OCS Front End when an RFC1006 transport is configured.

If you use an LTS_INET directive with explicit values for the IP_ADD and GW_ADD parameters, then this will automatically result in the IPS configuration to use the RFC1006 transport.

The following can be induced automatically:

```
IFCONFIG FD0 INET IPDSTAD @IP_ADD ARP TRAILERS          (*)
IFCONFIG LO0 INET IPDSTAD 127.0.0.1                  (****)
ROUTE ADD NET DEFAULT @GW_ADD 1                        (**)
ROUTE ADD HOST @IP_ADD @GW_ADD METRIC                 (***)
```

(*) if the IP_ADD parameter is given explicitly in the LTS_INET directive,

(**) if the GW_ADD parameter is given explicitly in the LTS_INET directive,

(***) one route for each INET_RT and NRTS (with TYPE = INET) when IP_ADD, GW_ADD, and METRIC parameters are given explicitly (in NRTS and INET_RT).

(****) if the LTS_INET directive is present.

Directives given explicitly in the IPS Configuration Subfile take precedence over the induced directives. If the IPS Configuration Subfile exists, then the IP address of FD0 must be the same as the IP_ADD value of the LTS_INET directive (if given).

8.3.11 FILE Directive

Purpose

This directive defines the name of subfile (a secondary file) which contains OSI/DIWS configuration records. The STACK, TCIVMO, RIB, and NLOC directives may be placed in this subfile. The NRSC, NRTS, NSDOM, LTS_INET, INET_RT, and FILE directives cannot be placed in this subfile.

The records contained in this subfile are processed in the same way as those of the OCS Front End specific subfile, except the NAME or IDENT parameter. In these records the NAME or IDENT parameter is a string of up to 7 characters, to which the configuration process adds a one character suffix specific to the OCS Front End (i.e index).

This directive is optional.

Syntax

```
FILE = <subfile_name>
```

Parameter Description

<subfile_name>	the name of a user's subfile in the library SYS.DSACONF.

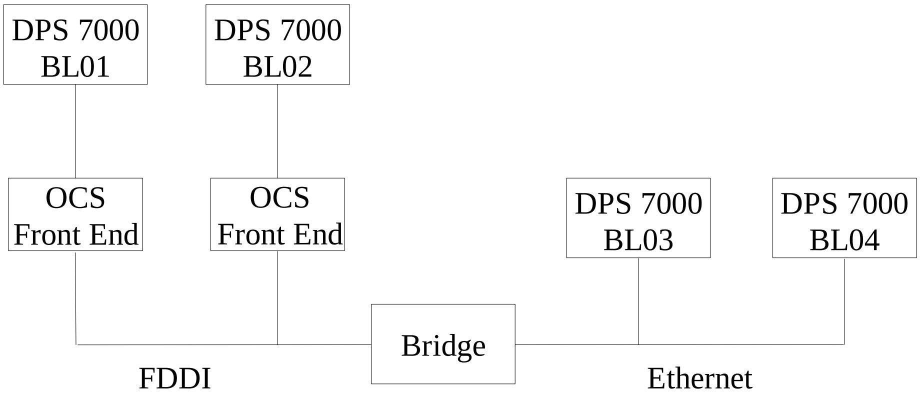
8.3.12 OCS Front End Configuration Description Examples

8.3.12.1 Example 1

I- NETWORK SCHEMATIC DESCRIPTION

This example shows:

- the local system BL01
- its neighbor systems BL02, BL03, and BL04.



BL02 MAC address: 080038100012
BL03 MAC address: 080038320001
BL04 MAC address: 080038310001

II DSA NETWORK CONFIGURATION DESCRIPTION

The OCS Front End Configuration Subfile (OSI/DIWS) of BL01 contains:

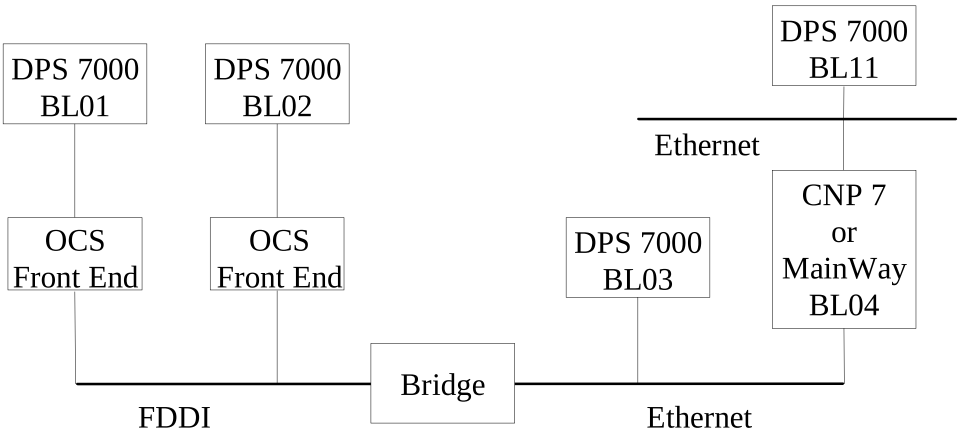
```
STACK TSNAME=EA01
TCIVMO NAME=IVM01 TPUSIZE=1024
TCIVMO NAME=IVM02 TPUSIZE=4096
NRSC NAME=BL02 NRTS=BL02
NRTS NAME=BL02 TYPE=DIWS MACADR=080038100012 IVMO=IVM02
NRSC NAME=BL03 NRTS=BL03
NRTS NAME=BL03 TYPE=DIWS MACADR=080038320001 IVMO=IVM01
NRSC NAME=BL04 NRTS=BL04
NRTS NAME=BL04 TYPE=DIWS MACADR=080038310001 IVMO=IVM01
```

8.3.12.2 Example 2

I- NETWORK SCHEMATIC DESCRIPTION

This example shows:

- the local system BL01
- its neighbor systems BL02, BL03, and BL04 (pass-through system).



BL01 MAC address: 080038100011
BL02 MAC address: 080038100012
BL03 MAC address: 080038200003
BL04 MAC address: 080038200004
BL11 MAC address: 080038200001

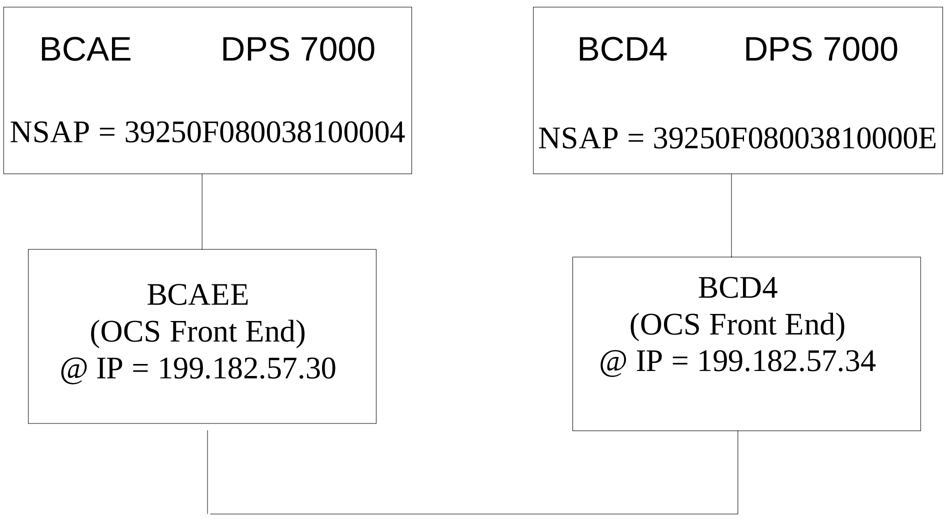
Note that ISO Transport is forbidden from BL01/BL02 to BL03 because MainWay/Microfep/Datanet must be in front of BL03.

II ISO NETWORK CONFIGURATION DESCRIPTION

The OCS Front End Configuration Subfile (OSI/DIWS) of BL01 contains:

```
STACK TSNAME=EA01
TCIVMO NAME=IVM01 TPUSIZE=1024
TCIVMO NAME=IVM02 TPUSIZE=4096
NRSC NAME=BL04 NRTS=BL04
NRTS NAME=BL04 TYPE=DSA57 MACADR=080038200004 IVMO=IVM02

NLOC NAME=BL01 LNSAP=39250F080038100011
RIB IDENT=BL02 RNSAP=39250F080038100012 IVMO=IVM02
                                        SNPA=080038100012
NSDOM NAME=NET1 NSAP_PRF=39250F08003820 NRTS=BL04
RIB IDENT=BL11 RNSAP=39250F080038200001 IVMO=IVM01
                                        SNPA=080038200004
```

8.3.12.3 RFC1006 Configuration Subfile Example



Configuration Subfile of OCS Front End BCAEE in BCAE host

This OCS Front End is used to establish RFC1006 communications from BCAE to BCD4.

```
LTS_INET NAME=LTSBCAEE IP_ADD=BCAEE PORT=102
                       MAXTC=1050 TPUSIZE=4096
# FOR DIWS APPLICATIONS
NRSC NAME=BCD4 NRTS=BCD4NCC1
NRTS NAME=BCD4NCC1 TYPE=INET IP_ADD=BCD4 PORT=102
# FOR OSI APPLICATIONS
INET_RT RNSAP=39250F08003810001E IP_ADD=BCD4 PORT=102
```

Configuration Subfile of OCS Front End BCD4 in BCD4 host

```
LTS_INET NAME=LTSBCD4 IP_ADD=199.182.57.34 MAXTC=1030 PORT=102
# FOR DIWS APPLICATIONS
NRSC NAME=BCAE NRTS=BCAENCC1
NRTS NAME=BCAENCC1 TYPE=INET IP_ADD=199.182.57.30
# FOR OSI APPLICATIONS
INET_RT RNSAP=39250F080038100014 IP_ADD=199.182.57.30
```

Example of ETC_HOSTS

```
199.182.57.30     BCAEE  bcaee  # this example BCAE Front End
199.182.57.31     BCAEF  bcaef  # another front end of BCAE
199.182.57.34     BCD4          # BCD4 front end of BCD4 host
```

8.4 Incremental Configuration

8.4.1 Purpose

Incremental configuration allows you to add new remote systems and new route descriptions to the current OCS Front End configuration.

Incremental configuration can only be used to extend the OSI/DIWS configuration.

It completes the dynamic configuration obtained by submitted OCS Front End administrative commands such as CRRB (Create RIB entry), CRAL (Create Addressing List) or IVMO management commands (MDIVMO, CRIVMO, ATTIVMO).

8.4.2 Definition

The incremental configuration subfile contains configuration directives to be added to the current OCS Front End main configuration ones. These directives allow new declarations of remote OSI/DIWS systems and/or new routes to reach remote systems. They cannot replace previous configuration directives read in the main configuration subfile.

If no current OSI/DIWS configuration has been performed, no incremental configuration can be executed.

Incremental configuration only allows you to add a new route to a remote system if the Transport Station (COTP, DIWS, INET) used to reach it has been configured. For example: an LTS_INET directive is needed in the main configuration subfile for the INET Transport Station.

8.4.3 Use

8.4.3.1 Automatic Start

During the start of the OCS server, after successful analysis of the main OSI/DIWS configuration subfile of the front end, an incremental configuration step is started.

In that case an incremental configuration subfile is analysed. This subfile, EAxx_ISOCONF_INCR (where EAxx equals the current OCS Front End name), is held in the SYS.DSACONF library.

The result of the analysis is appended to the EAxx_CONF_RPRT report subfile.

8.4.3.2 Operator Start

The operator can start an incremental configuration at any time while the OCS server attached to the front end is running an OSI/DIWS configuration.

The UPGRADE_FCP_CONFIG operator command is used to submit an incremental configuration to the current OCS Front End configuration. In this case, subfile name and library name are chosen by the user.

The result of the analysis is appended to the EAxx_CONF_RPRT report subfile (in SYS.DSACONF library).

If the same unmodified incremental subfile is submitted two (or more) times the second (and subsequent) attempts are ignored and a warning is written to the report subfile.

8.4.4 Permitted Directives

The following directives are allowed in the incremental configuration subfile under the indicated conditions:

NSDOM **NRSC**	With a new NAME, it means creation of a new NRSC or NSDOM entry. With a name already specified in the main configuration subfile and with a new NRTS list, it means the addition of new routes to reach this NRSC or NSDOM. The directive fails in case of duplicate association (same NAME and same NRTS).
TCIVMO **RIB** **NLOC**	Only if OSI/DIWS protocol has been configured in the main configuration file. Functions only with a new NAME (or IDENT). The directive fails in the case of a duplicate TCIVMO, RIB or NLOC name for the same OCS Front End.
NRTS with TYPE=DSA57 or TYPE=DIWS	Only if OSI/DIWS protocol has been configured in main configuration file. Functions only with a new NAME. The directive fails in the case of a duplicate NRTS name for the current OCS Front End.
NRTS with TYPE=INET	Only if an INET TS has been configured in the main configuration subfile. Functions only with a new NAME. The directive fails in case of duplicate NRTS name for the current OCS Front End.
INET_RT	Only if an INET TS has been configured in the main configuration subfile. Functions only with a new association (RNSAP,@IP). The directive fails in case of duplicate association (same RNSAP and same @IP).

The STACK, LTS_INET and FILE directives are not allowed in an incremental configuration subfile.

8.5 IPS Configuration

8.5.1 IPS Configuration System Subfiles

Several system subfiles are used during the IPS configuration. These subfiles are located in the GCOS 7 library named SYS.DSACONF.

The subfiles are:

- IPS Configuration Subfile
- ETC_HOSTS Subfile
- ETC_NETWORKS Subfile
- ARPCO_name Subfile

8.5.2 IPS Configuration Subfile

Name

The IPS Configuration Subfile is a subfile of the GCOS 7 library named SYS.DSACONF. The name of the IPS Configuration Subfile is given in GO order of the scenario subfile.

There is only one such subfile per OCS Front End connected to the DPS 7000 local host.

This subfile is referenced at STSVR time for the server of the OCS Front End to be configured.

Structure

This subfile contains a list of configuration command lines. Command lines are character strings made of printable characters where either upper or lower case can be used.

Command lines are separated from each other by means of the new line character. The subfile may contain blank lines or comment lines. A comment line must begin with the "#" character. Command name and arguments or parameters are separated by any number of blank characters.

The total length of a command line should not be greater than 255 characters including the new line character.

Contents

This subfile contains the commands needed to configure the IPS stack and to initialize the physical interfaces. The subfile must contain at least one IFCONFIG command needed to configure the OCS Front End physical interface (fd0 or lo0). It may also contain the optional ARP and ROUTE configuration commands. In this case, before the first ARP or ROUTE configuration command, the IFCONFIG configuration command is mandatroy for the fd0 interface.

The 3 commands (to configure the IPS stack) are:

- IFCONFIG
- ROUTE
- ARP

Note that these commands can also be executed at other times when network administrator decides to get and/or modify some network parameters (see the *Network User Guide*).

8.5.3 ETC_HOSTS Subfile

Name

This is a subfile of the GCOS 7 library named SYS.DSACONF. The name of the subfile is ETC_HOSTS.

This subfile contains the symbolic names of the hosts connected to the network (including the local host) and their corresponding addresses (Internet address or NSAP).

The subfile is used to get the address of a host when a command makes a reference to the symbolic name of this host.

There is only one such subfile per GCOS 7 system.

Access to the ETC_HOSTS subfile is performed by both IPS and OSI/DIWS administrative and configuration commands (described in this paragraph and the preceding one), or by primitives defined in the GXTI Name Service.

Note that OPEN 7 has its own separate, equivalent ETC_HOSTS subfile.

Structure

This subfile is made of hosts lines. It can also contain blank lines or comments lines.

Lines are separated from each other by the new line character.

Upper or lower case characters can be used when writing command lines. The length of each host line should not exceed 255 characters including the new line character.

Each host line is composed of fields separated by any number of blank characters, containing the following information:

- address,
- official host name,
- aliases,
- comments.

Contents

The contents of each line is as shown below:

```
Host_address  Host_name  [alias][alias]  comment
```

where:

Host_address	- IP address (in 4 part decimal dot notation) when the host is connected on a TCP/IP network, or - NSAP list (in hexadecimal format prefixed by '@iso') when the host is connected on an OSI network.
Host name, **alias**	is the symbolic name of the host (up to 15 characters).
comment	is an optional string. A comment line or comment field must begin with the "#" character.

EXAMPLE:

```
# ETC_HOSTS file
# --------------
#
# Internet Address Examples:
#
199.182.51.8  host1 HOST1 # IP addr: 81b63308 hexa
# The previous line is an Internet host description that
# contains:
#   - the Internet address [199.182.51.8], dot notation for
#     81b63308 hexa
#   - the official name    [host1]
#   - an alias list        [HOST1]
#   - a comment            [# IP addr: 81b63308 hexa]
#
141.112.1.10   host2      # IP addr: 8d70010a hexa
# The previous line is an Internet host description that
# contains:
#   - the Internet address [141.112.1.10], dot notation for
#     8d70010a hexa
#   - the official name    [host2]
#   - no alias
#   - a comment            [# IP addr: 8d70010a hexa]
#
# ISO Network Service Point Examples:
#
@iso,3802c4d6   host1   host3   HOST3   host_3   HOST_3
# The previous line is an ISO host description that
# contains:
#   - an NSAP              [3802c4d6]
#   - the official name    [host1]
#   - an alias list        [host3  HOST3  host_3  HOST_3]
#   - no comment
#
@iso,524,72f4,a007d host4   # nsaps: 524 or 72f4 or a007d
# The previous line is an ISO host description that
# contains:
#   - several NSAPs        [524,72f4,a007d]
#   - the official name    [host4]
#   - no alias
#   - a comment            [# nsaps: 524 or 72f4 or a007d]
#
#
#
# localhost: loopback driver
127.0.0.1   localhost local
```

❑

8.5.4 ETC_NETWORKS Subfile

Name

This subfile is a subfile of the GCOS 7 library named SYS.DSACONF. The name of the subfile is ETC_NETWORKS.

This subfile contains the symbolic names of each single network the local host can access.

As for host names, networks can be referred to by their symbolic name. The ETC_NETWORKS subfile is used to get the corresponding IP address.

There is only one such a subfile per GCOS 7 system.

Access to the ETC_NETWORKS subfile is performed by the administrative (described in the Network User Guide) and the configuration commands described below.

Note that OPEN 7 has its own separate, equivalent ETC_NETWORKS subfile.

Structure

This subfile has about the same structure as the ETC_HOSTS subfile except address and name fields are inverted as shown in "Contents" below.

Contents

```
Network_name  Network_address  comments
```

where:

Network_name is the symbolic name of the network (up to 15 characters).

Network_address is the IP address of the network written as an up to four parts dot notation address (see "IPS Notation" later in this chapter).

8.5.5 ARPCO Subfile

There is an additional configuration ARP subfile which can be referred to by the ARP -f command described below. This subfile contains a list of ARP entry creation commands (which can be common to several OCS Front Ends).

Name

The name of this type of subfile is:

```
ARPCO_name
```

where, ARPCO_ is a mandatory prefix, and name is a string chosen to complete the subfile name.

Structure

This subfile contains a list of ARP commands. Command lines are separated from each other by means of the new line character.

Contents

The ARP commands are structured exactly as the ARP -s (described in paragraph *Internet Address Notation* later in this chapter), except that the ARP command name and the -s option are omitted in each command line.

8.5.6 Maintaining the IPS Configuration Subfiles

The IPS Configuration Subfiles above (except the report subfile) can be created using any GCOS 7 editor. All can be displayed and/or modified the same way.

Dynamic modifications of ETC_HOSTS and ETC_NETWORKS subfiles are applied to these subfiles immediately. Modifications of IPS configuration and ARPCO type subfiles will be applied at next OCS server restart.

To make it easier for you to create the Configuration Subfile, two GCL commands are available in the IOF domain. The GCL command BUILD_FCP_CONFIG builds the Configuration Subfile. The GCL command VERIFY_FCP_CONFIG performs a first level of command syntax verification of the Configuration Subfile. See "OCS Front End Configuration Tools" later in this chapter.

8.5.7 SNMP Agent Configuration Subfiles

Note: SNMP is only available on Bull hardware OCS Front End.

For OCS Front End, an SNMP agent is located in the OCS server. This allows remote administration of the IPS stack and the FDDI physical access point.

The SNMP agent needs the following configuration subfiles:

- <LCT_name>_SNMP_CONF
- <LCT_name>_SNMP_COMM
- <LCT_name>_SNMP_TRAP
- <LCT_name>_SNMP_AUTH

where LCT_NAME identifies the OCS Front End to which the SNMP agent is attached.

These are subfiles of the library SYS.DSACONF.

8.5.7.1 Configuration Subfile <LCT_name>_SNMP_CONF

This subfile is optional for an active SNMP agent. If it is missing, default values apply.

This subfile is used to initialize information of the system group (of the MIB of the local agent). So all of the elements of this group are READ-ONLY when interrogated by a manager. (This over-rides the access mode given in *RFC1213* for the objects sysname, syscontact and syslocation.) This group contains information specific to the agent.

This subfile also contains optional configuration information that does not belong to the system group (port numbers).

Each line of the <LCT_name>_SNMP_CONF subfile contains the following 2 items separated by an equal (=) sign:

- a keyword,
  which identifies the element. The keyword may be in upper or lower case,
- a value,
  which is the rest of the line (including blanks).

Each line is up to 255 characters long.

The elements described in the <LCT_name>_SNMP_CONF subfile are:

- sysdescr
  This describes the hardware/software combination of the managed entity. The keyword for this element is DESCR. The default value is blank.
  For example, DESCR = NCC3.1/V4.2, describes an OCS Front End whose hardware version is 3.1 and whose software version is V4.2.

- sysobjectid
  This identifies the network subsystem contained in the managed entity. This is a means of determining what kind of "box" is being managed. The keyword for this element is OBJID. The value must be an object identifier contained in the MIB. The default value is OBJID = Bull. For example, OBJID = BullFcp, identifies the agent Bull for an OCS Front End.

- sysname
  This is the name of the OCS controller. The keyword for this element is NAME. The OCS Front End name is of the form <host_name>_<LCT_name>. The default value is blank.
  For example, NAME = BC08_EA02, describes the OCS controller named EA02 which is attached to the host named BC08.

- syscontact
  This identifies the contact person for the node. The keyword for this element is CONTACT. The default value is blank.
  For example, CONTACT = John Smith, tel: 12345, gives the name and telephone number of the contact person.

- syslocation
  This identifies the physical location of the node. The keyword for this element is LOCATION. The default value is blank.
  For example, LOCATION = CL B1/1515, identifies the location.

- sysservices
  This identifies the services offered by the entity. The keyword for this element is SERVICES. SERVICES is a decimal number which is the sum of 2**(layer - 1) for each layer for which services are offered. The default value is SERVICES = 15, which means layers 1, 2, 3, and 4, since 15 = 2**(4-1) + 2**(3-1) + 2**(2-1) + 2**(1-1).
  For example, SERVICES = 72 means layers 4 and 7, since 72 = 2**(7-1) + 2**(4-1).

- port number (decimal value) (1 ó port ó 65535)
  This identifies the UDP port used for normal communication. The keyword for this element is PORT. The default value is PORT = 161.
  For example, PORT = 1161, means port number 1161 is used for normal communication.

- trap port number (decimal value) (1 ó trap port ó 65535)
  This identifies the UDP port used for trap messages. The keyword for this element is TRAPPORT. The default value is TRAPPORT = 162.
  For example, TRAPPORT = 1162, means port number 1162 is used for trap messages.

8.5.7.2 Community Configuration Subfile <LCT_name>_SNMP_COMM

This subfile is mandatory for an active SNMP agent.

Access to the agent is controlled by the SNMP community.

The subfile <LCT_name>_SNMP_COMM contains the community list used by the agent. Access to the agent can be restricted using SNMP communities. For OCS Front End, at least one manager station must be described in the subfile <LCT_name>_SNMP_COMM.

Each line of the <LCT_name>_SNMP_COMM subfile consists of 3 items separated by one or more blanks.

The items are:

- a community name which identifies the Manager Station,
- the IP address (in dot notation) of the remote site,
- the privileges (READ, WRITE, NONE) accorded to the remote manager.

The privilege NONE can be used to lock out a specific community.

An example of a line of the <LCT_name>_SNMP_COMM subfile is:

```
lautrec        129.181.16.213        WRITE
```

which means that the manager at the IP address 129.181.16.213, which is associated with the community name "lautrec", has the WRITE privilege on the agent.

8.5.7.3 Trap Community Configuration Subfile <LCT_name>_SNMP_TRAP

This subfile is optional for an active SNMP agent.

An agent can send trap messages to signal the occurrence of certain events.

The OCS Front End agent can generate the following 3 types of trap message:

- coldStart type at agent launching (start of OCS server),
- warmStart type at agent restarting,
- authentification failure type when the agent detects an illegal access from a manager (either not declared in the <LCT_name>_SNMP_COMM subfile, or access denied).

The <LCT_name>_SNMP_TRAP subfile contains a list of the systems that should be notified when a trap event occurs. Generally, only management stations should be included in this subfile.

Each line of the <LCT_name>_SNMP_TRAP subfile consists of 3 items separated by one or more blanks.

The items are:

- a community name which identifies the community,
- the IP address (in dot notation) of the remote site,
- the UDP port (the default port number is 162) to which the trap messages are to be sent.

An example of a line of the <LCT_name>_SNMP_TRAP subfile is:

```
ravel        129.181.113.132        162
```

which means that the trap messages are sent through the port number 162, to the system at the IP address 129.181.113.132, using the community name "ravel".

8.5.7.4 Authentication Failure Configuration Subfile <LCT_name>_SNMP_AUTH

This subfile is optional for an active SNMP agent.

This subfile allows you to enable/disable the sending of authentication failure trap.

It has only one line consisting of 2 items separated by an equal sign (=):

- the keyword: authenticationtrap
- the value: enabled or disabled

For example,

```
authenticationtrap = enabled
```

```
default value: enabled
```

8.5.8 SNMP Manager Configuration Files

Knowledge of SNMP and MIB definitions is required before reading this paragraph. If necessary, refer to the published Internet community standards.

Before using the manager, you must have configured at least the following subfile: SYS.DSACONF..SMGR_COMMPORT.

8.5.8.1 SYS.DSACONF..SMGR_COMMPORT Subfile

This subfile is mandatory. Its purpose is to associate a port number and a community to a host identification. Consequently, when you enter an SNMP manager command, you just give the host id and the corresponding port and community are found from this subfile.

Each line is composed of 3 fields (separated by a blank):

- **A Host Identification.**
  This can be either an IP address (in decimal dot notation) or a symbolic name (which must be associated with an IP address in the SYS.DSACONF..ETC_HOSTS subfile). In particular, the host identification may be one of the local OCS Front Ends. The value you give in the HOST keyword of each manager command must be one these host identifications. When you enter a command, you omit the port number and community name as they are retrieved via HOST from the ETC_HOSTS subfile.
- **A Port Number.**
  This is the UDP port number of the SNMP agent which is located on the host identified by the preceding field. It is expressed as a decimal number.
- **A Community Name.**
  This must correspond to the community contained in the community file of the agent concerned. In particular, for a local OCS Front End agent, the community name must be declared in the corresponding SYS.DSACONF..EAxx_SNMP_COMM subfile (see the previous paragraph).

EXAMPLES:

```
129.183.53.35 161 BCC1NCC2
BCC1-NCC1 161 NCC1
```
❑

The first example shows the host identification as an IP address (129.183.53.35) with a port number (161) and a community name (BCC1NCC2).

The second example shows the host identification as a symbolic name (BCC1-NCC1) with a port number (161) and a community name (NCC1).

8.5.8.2 SYS.DSACONF..SNMP_MGR_MIB Subfile

This subfile is optional. Its purpose is to add new entries to the MIB table.

The MIB table gives the correspondence between object identifiers (for example, 1.3.6.1.2.1.2.2.1.4) and mnemonics (for example, ifMtu). Consequently, when you enter a command, you can use the mnemonic instead of the identifier (the correspondence is found from this MIB table).

There are many pre-defined correspondences in the MIB table: they describe the objects in RFC1213 (mib 2), RFC1285 (fddi mib), and the kernel objects of OCS Front End. To see the pre-defined entries, refer to the RFC1213 or RFC1285 documentation, or execute an snmpnext command against the whole MIB of a local agent (if such an agent is operational). See the examples of the snmpnext command in the *Network User Guide*.

Each line is composed of 2 fields (separated by a blank):

- **An Object Identification.**
  This is expressed in decimal dot notation (up to 50 decimal numbers, if each number is less than 127).
- **An Object Name.**
  This is a character string to be used as a mnemonic (up to 64 characters).

Lines which begin with a "#" character are treated as comments.

Each line of SYS.DSACONF..SNMP_MGR_MIB is handled as follows:

1. If the Object Identifier in the 1st field already exists in the predefined MIB table, the mnemonic in the 2nd field replaces the existing mnemonic (in the current MIB table).
2. If the Object Identifier in the 1st field does not already exist in the predefined MIB table, the correspondence "object identifier - mnemonic" is added to the current MIB table.

Beware of duplicate mnemonics (that is, the same mnemonic associated with 2 or more object identifiers). In such a case, when the mnemonic is used in a command, it is always replaced by the first corresponding object identifier found in the current MIB table.

EXAMPLE:

```
1.3.6.1.8.12.24 truc
```
❑

This example adds a correspondence between an object identifier (1.3.6.1.8.12.24) and a mnemonic (truc).

8.5.8.3 SYS.DSASLIB..NCC_MG_DOCFILES Subfile

This subfile is optional. It contains a list of names of documentation files.

Each line contains one documentation file name.

If this subfile does not exist, then the following documentation file is used by default:

```
SYS.DSASLIB..NCC_MG_RFC1213
```

Currently, SYS.DSASLIB..NCC_MG_DOCFILES contains 3 names (see the next paragraph).

8.5.8.4 The SNMP Objects Documentation Files

The documentation files contain data to be displayed by the SNMPDOC command (see the *Network User Guide*). The names of these files are contained in the SYS.DSASLIB..NCC_MG_DOCFILES subfile (see the previous paragraph).

Currently, there are 3 documentation files:

```
SYS.DSASLIB..NCC_MG_RFC1213
SYS.DSASLIB..NCC_MG_RFC1285
SYS.DSASLIB..NCC_MG_FCP_MIB
```

You may add new documentation files. If you do, then:

- Their format must be compliant with the SYS.DSASLIB..NCC_MG_RFC1213 file.
- You must add their names in the SYS.DSASLIB..NCC_MG_DOCFILES subfile.

A specific command enables you to create the new documentation files in compliance with the SNMPDOC command.

Note that at each delivery of OCS Front End software, the following 4 subfiles are re-installed:

```
SYS.DSASLIB..NCC_MG_DOCFILES
SYS.DSASLIB..NCC_MG_RFC1213
SYS.DSASLIB..NCC_MG_RFC1285
SYS.DSASLIB..NCC_MG_FCP_MIB
```

8.6 IPS Notation

8.6.1 IPS Command Syntax Notation

A command line is made of a command name followed by an argument and parameter list.

The parameters list is made of positional parameters,keywords and Self Identifying Values (SIV):

- Positional parameters must appear first just after the command name.
- Keywords and SIV follow them and can be placed in any order.

ITEM	When written in upper case ITEM means it is a keyword that must appear as it is specified.
item	When written in lower case item means it is a parameter value to be provided by the submitter of the command.
[item]	When located between brackets, item is an optional parameter that can be omitted.
{item1 | item2 | <u>item3</u>}	When several items are located between braces this means a choice can be made among the proposed items. An underlined item indicates the item default value.

8.6.2 Internet Address Notation

An IP address consists of a string representing the address in dot notation.

Dot notation consists of up to four decimal numbers separated by a dot.

Each decimal part is interpreted as one byte of hexadecimal data which is assigned to each of the four bytes of the Internet address from left to right.

If less than four parts are provided remaining bytes into the Internet resulting address are set to 0.

A decimal part should not have a value greater than 255.

There must be no more than 3 leading 0's in each part of the address.

There must be no blank character inside the IP string.

8.6.3 MAC Address Notation

A Medium Access control address consists of a string made of twelve hexadecimal digits arranged in pairs separated by the colon character.
There must be no blank character inside the MAC string.

8.7 IPS Configuration Commands

8.7.1 IFCONFIG

Purpose

During configuration, this command is used to assign Internet addresses to the FDDI and LOOP OCS Front End interfaces and to set up some options related to these interfaces. Addresses and options are set independently for each interface according to the parameters of the command.

This command is mandatory.

Syntax

```
IFCONFIG  interface_name

          address_family

          IPDSTAD dest_add

          [parameters]
```

The following can appear within [parameters]:

```
          BROADCAST broadcast_add
          NETMASK mask
          {arp | noarp}
          {trailers | notrailers}
```

Parameter Description

Interface_name	is a 3 characters positional argument which indicates the name of the interface the command applies to. Interface_name must have the following values: – fd0 for the FDDI interface. – lo0 for the loopback interface LOOP.
address_family	is a positional argument identifying the protocol the command applies to. Address_family will always be represented as a string equal to INET.

IPDSTAD	is a keyword argument which allows to identify the following statement as being the dest_add definition.
dest_add	is the internet address to be assigned to the OCS Front End interface interface_name. It is expressed either as a host name (which must be present in the ETC_HOSTS member of SYS.DSACONF) or as an IP address in decimal dot notation (see "IPS Notation" in this chapter).
BROADCAST	is a keyword which precedes the broadcast_add definition. BROADCAST is not allowed with the lo0 interface.
broadcast_add	is an IP address expressed in a four decimal part dot notation. It specifies the address to use when it is expected that several systems on the network receive the same message. It is made of the host part of the address set to 1's.
NETMASK	is a keyword which precedes the mask definition.
mask	is a string of 8 hexadecimal digits. It is used to subdivide the host part of an internet address into subnetworks. This is expressed by a bit mask using 1's for the bits tied to network and subnet number and 0's for the bits used to address hosts on the subnet.
arp	enables the use of the address resolution protocol to dynamically map between internet and ethernet address and to cache the mapping. This is a SIV parameter; arp is the default value. Note that arp is not allowed with the lo0 interface.
noarp	disables the address resolution protocol. It is used when a host on the network does not implement the protocol or if it is expected not to dynamically modify the cache namely when it is manually configured (see ARP command). This is a SIV parameter.

trailers requires the use of a link level encapsulation when sending messages to facilitate memory management at receiving host level. That is to say variable parts in length of a data header are moved at the end of the data to insure page alignment in a page mapped virtual memory management.
Trailers is a SIV parameter.
Trailers is the default value.
Trailers is not presently used by OCS Front End. It is reserved for future use. Nevertheless, the trailers value is not allowed for lo0 interface and generates an error (if used).

notrailers disables the use of the trailers encapsulation.
Notrailers is not presently used by OCS Front End. It is reserved for future use.

Command Execution

- The command line is analyzed. Arguments are checked for type range and value.
- If an error is detected, it is noted and processing continues or stops according of the error severity (warning or fatal error).
- Else a command record is built according to the arguments and parameters of the command and it is sent from the local DPS 7000 host to the OCS Front End concerned. Processing waits for the response it is expected to get.
- The response is analyzed and processing resumes or it is aborted depending of the result indicated by the response.

Error Detection

- A syntactical or a semantic error is found during command analysis.
- No response occurs within a required time.
- The response indicates the OCS Front End concerned has detected an error during execution of the command.
- The response is not the expected one. That is because some items inside the response does not correspond to those of the request (request id, OCS Front End id, function).
- All error messages (whether fatal or not) are recorded in the subfile EAnn_CONF_RPRT attached to the server. A message is also displayed on the operator console. This messages states that there are configuration errors/warnings and asks the operator to read the report subfile. In addition, a fatal error stops the processing.

8.7.2 ARP

Purpose

During configuration, this command is used in several ways depending on an ARP command option. Two possibilities are offered. They are:

- Individual creation of an ARP entry each time an ARP command is submitted for execution in the subfile.
- Global creation of entries by means of a subfile containing several ARP commands.

Each command creates an Address Resolution Protocol entry into the ARP translation table managed by the IPS stack into the OCS Front End.

Each entry in this table caches an Internet/Ethernet address mapping set according to the arguments of the command.

All the entries created during configuration process are permanent entries.

According to the published option, a OCS Front End may act as an ARP server responding to requests coming from another non ARP machine, for the entries declared published in the table. This allows a OCS Front End configured manually to pass its ARP information to the other ones.

The ARP command is optional for the configuration.

Syntax (Individual ARP entry creation)

```
ARP -s  host_name | host_ipadd

        mac_add

        [published]

        [trailers]
```

Parameter Description

-s	a positional string which indicates that the ARP command is being used to create an entry in the ARP table.
host_name	symbolic name of the host. It is used to look for the corresponding internet address into the ETC_HOSTS subfile. This Internet address is equivalent to host_ipadd described below. Host name is a string made of up to 15 printable characters except blank, dot and "#" characters. Host_name or host_ipadd (whichever one you choose) is a positional parameter.
host_ipadd	host IP address, when directly provided by the command. Host IP address is expressed in dot notation as a four parts address. Host_ipadd or host_name (whichever one you choose) is a positional parameter.
mac_add	physical address which is associated to the host_ipadd. mac_add is expressed in MAC address notation as a six hexadecimal parts address. It is a positional argument.
published	makes the created ARP information to be known by another host. This is a SIV parameter. There is no default value.
trailers	requires the trailers encapsulation (see IFCONFIG command). This is a SIV parameter. There is no default value.

Command Execution

- The command line is analyzed and arguments are checked for validity. If a host name is provided by the command, an access to the ETC_HOSTS subfile is done to get the corresponding IP address.
- A record is built according to arguments and options of the command and it is sent to the OCS Front End concerned.
- The incoming response is checked and processing resumes if no error is detected.

Error Detection

- A syntactical or a semantic error is found during command analysis.
- host_name is not found while scanning the ETC_HOSTS subfile.
- No response is provided within a required time.
- The response indicates an error.
- All error messages (whether fatal or not) are recorded in the subfile EAnn_CONF_RPRT. In addition, a fatal error stops the processing.

Syntax (Global ARP entries creation)

```
ARP  -f

     ARPCO_name
```

Parameter Description

-f	a positional string which indicates that the subfile ARPCO_name is to be accessed.
ARPCO_name	name of the subfile containing the ARP commands to be performed.

Command Execution

- ARP commands are taken one by one from the subfile.
- Each time a command is taken from the subfile, a record is sent to the OCS Front End concerned and the corresponding expected response is checked.
- Processing continues until all commands are processed even if an error is detected.

Error Detection

- The subfile cannot be opened or contains an erroneous or an illegal command.
- File_name does not respect the ARPCO_ naming convention.
- The issued response indicates an error or there is no response from the OCS Front End.
- All error messages (whether fatal or not) are recorded in the subfile EAnn_CONF_RPRT. In addition, a fatal error stops the processing.

8.7.3 ROUTE

Purpose

During configuration, this command is used to manipulate the routing tables managed by the IPS stack in the OCS Front End. The command adds a route in an entry of the table.

Each route indicates the destination address to be reached. If the destination is not on the same network as the local host, the route gives the gateway address to pass through. If the destination address is on the same network as the local host, the gateway address given is the local host IP address.

There are as many ROUTE commands in the IPS configuration subfile as there are networks, subnetworks, hosts destinations to reach.

This command is optional for the configuration.

Syntax

```
ROUTE add  NET | HOST

           name | ip_add

           gateway_name | gateway_ipadd

           metric
```

Parameter Description

add This 3 character string indicates that the ROUTE command is used to add a route into the IPS stack routing table. It is a positional argument.

NET | HOST These two positional keywords indicates how the next part (name or ip_add) of the address parameter is to be interpreted.

If NET, the next part of the address parameter is interpreted as a network address.

If HOST, the next part of the address parameter is interpreted as a host address.

name | ip_add

This parameter is either the symbolic name of the destination to reach or its IP Internet address.

If a symbolic name is provided, the corresponding IP address is taken either from the ETC_NETWORKS or from the ETC_HOSTS subfile depending on whether the preceding parameter NET or HOST (respectively). Otherwise, the IP address is directly taken from the command line.

Name is a string of up to 15 printable characters. String should neither contain nor blank, nor dot nor "#" characters.

A host address is expressed as a four decimal parts dot notation address when up to four parts are used to express a network address, missing parts being filled with 0's on the right during command processing.
When Default is given to this parameter, this indicates a default route declaration. This route will be used for outgoing packets when no entry matches in the OCS Front End IP routing table (useful with NET parameter).
Default does not need to be declared in the ETC_HOSTS or ETC_NETWORKS subfile.

gateway_name | gateway_ipadd

This parameter is either the symbolic name of the gateway or its IP address.

If a name is provided, the corresponding IP address is fetched from the ETC_HOSTS subfile else it is directly taken from the command line.

Name is a up to 15 printable characters string. String should neither contain nor blank, nor dot nor "#" characters.
In every case, the IP address is expressed as a four decimal parts dot notation address.

metric

This mandatory parameter indicates the number of hops to the destination.It is expressed as an up to nine decimal digits.

Command Execution

- The command line is analyzed and arguments are checked for validity.
- A record is built according to arguments and options of the command and it is sent to the OCS Front End concerned.
- The incoming response is checked and processing resumes if no error is detected.

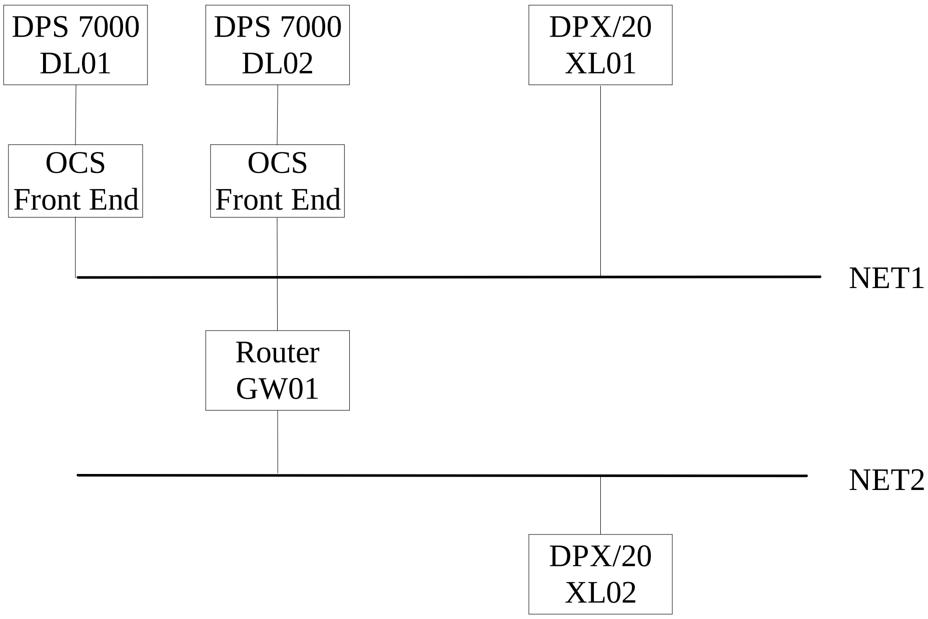
Error Detection

- A syntactic or a semantic error is found during command analysis.
- The host name or the gateway name is not found while looking up in the ETC_HOSTS subfile.
- The network name is not found while looking up in the ETC_NETWORKS subfile.
- No metric value is provided.
- No response is provided within a required time.
- The response coming from the OCS Front End indicates an error.
- All error messages (whether a fatal error or not) are recorded in the subfile EAnn_CONF_RPRT. In addition, a fatal error stops the processing.

8.7.4 Example (OCS using IPS)

This example describes the IPS configuration of the system named DL01.

The OCS Front End (on the left of the diagram) must be declared in the GCOS 7 NETGEN (via SVR and LCT directives) of the system DL01.



Address Definition

```
DL01 MAC address : 08:00:38:10:00:11
DL02 MAC address : 08:00:38:10:00:12
GW01 MAC address : 06:00:00:00:00:03
XL01 MAC address : 04:00:00:00:00:04

DL01 IP address : 193.182.55.1
DL02 IP address : 193.182.55.2
GW01 IP address : 193.182.55.3
XL01 IP address : 193.182.55.4
XL02 IP address : 193.182.56.4
```

IPS Configuration Subfile

```
      IFCONFIG fd0 INET IPDSTAD 193.182.55.1 NOARP
*     ROUTE add HOST DL02 DL01 0
*     ROUTE add HOST XL01 DL01 0
      ROUTE add HOST XL02 GW01 1
*     ROUTE add HOST GW01 DL01 0
      ROUTE add NET NET2 GW01 1
      ARP -s DL02 08:00:38:10:00:12
      ARP -s XL01 04:00:00:00:00:04
      ARP -s GW01 06:00:00:00:00:03
```

The directives marked with an asterisk (*) may be omitted. The IPS stack can determine the routing from the IP addresses.

ETC_HOSTS Subfile

```
193.182.55.1     DL01
193.182.55.2     DL02
193.182.55.3     GW01
193.182.55.4     XL01
193.182.56.4     XL02
```

ETC_NETWORKS Subfile

```
NET1  193.182.55
NET2  193.182.56
```

8.8 OCS Front End Configuration Tools

8.8.1 BUILD_FCP_CONFIG

Purpose

To facilitate OCS Front End configuration and to provide a first level of semantic check.

Syntax

```
BUILD_FCP_CONFIG  [ISO=Y/N]  [IPS=Y/N]
```

Parameter Description

ISO	An ISO/DIWS configuration subfile is to be generated. The configuration subfile can be the main file or a subsidiary file (see directive FILE).
IPS	An IPS configuration subfile is to be generated. The configuration subfile can be a main one, or a subsidiary ARPCO_name subfile.

Command Execution

The BUILD_FCP_CONFIG command displays a welcome screen of commands which allow you to create a whole ISO/DIWS or IPS configuration.

You cannot exit normally from the command without declaring the mandatory directives (these depend on whether the configuration is for IPS or ISO). You can force an abnormal exit.

BUILD_FCP_CONFIG does not work if the subfile already exists.

8.8.2 VERIFY_FCP_CONFIG

Purpose

To do a semantical and syntactical check of an ISO/DIWS or IPS configuration. This comand is useful if you modify the subfile using an editor (for example, FSE or EDIT).

To perform a coherence check of an ISO/DIWS or IPS configuration subfile.

The result of the verification is stored in a report subfile.

Syntax

```
VERIFY_FCP_CONFIG  TYPE=(ISO|IPS)
                   LIB=library_name
                   MEMBER=file_name
                   [UPGRADE=file_name]
                   REPORT = report_file_name
```

Parameter Description

TYPE	type of configuration (ISO/DIWS or IPS).
LIB	library (type SL) where the subfile to be checked is located.
MEMBER	member name which contains the configuration commands.
UPGRADE	member name which contains the incremental configuration directives. This member is located in the same library as MEMBER. Meaningless if TYPE=IPS.
REPORT	member name where configuration checking report will be done. The report subfile is located in the same library as MEMBER.

Command Execution

Each command line is analyzed according to the configuration type and copied in the report subfile.

If a warning condition or a fatal error is detected, a specific message is reported behind the command line report.

❑

9. Tuning the Network

9.1 Tuning FEPS, OCS, and TNS

The term *pool* refers to the global set of buffers assigned to the server for which the XBF statistics report applies.

The XBF statistics in the JORs of both TNS and FEPS shows the effects of BUFSZ, BUFNB, and EXT parameter values declared in the corresponding SVR directives, on the buffers of each server concerned at the time of:

- ***creation:***
  The buffer sizes are created solely on the basis of the values specified for BUFSZ and the maximum and minimum values for BUFNB.
- ***extension:***
  The buffers are increased to the maximum BUFNB when the percentage of used/allocated buffers is greater than or equal to the threshold defined by EXT.
- ***termination:***
  The session terminates and an account is given of the thresholds and the maximum utilisation, expressed as a percentage, of each individual buffer deleted.

For FEPS, additional information on the traffic handled by the Datanet(s) follows the XBF statistics.

9.1.1 Choosing BUFSZ and BUFNB Values

At network generation, it is recommended to use the default values, namely:

- BUFSZ = 128 bytes per unit
- BUFNB = 512 units (minimum) and 2048 units (maximum).

These values result in a total buffer pool size ranging from 64 K bytes (minimum) and 4 * 64 K bytes (maximum).

Any modification in BUFSZ and BUFNB can only be to scale up these values, since no value less than the default is recommended.

9.1.2 Choosing the EXT Value

At network generation, it is recommended to use the default, namely:

```
EXT = 80 % or 0.8 of the buffer space
```

For OCS, EXT = 80%, the default, is also the maximum value.

EXT determines the extension threshold of the buffer. This value can either be scaled up or down depending on how frequently the buffer is extended.

If the space occupancy of a buffer is equal to or greater than the extension threshold, 1 segment of 64k bytes (equivalent of 512 default buffer units) is allocated at a time. The space occupancy of the buffer is constantly monitored until the maximum number of buffers declared in BUFNB is reached.

EXT can be proportionally scaled up or down depending on the use of the buffers. This is done by using statistics over several sessions.

9.2 MAM Tuning

The information used for tuning the sizes of queues is to be found in:

- Appendix A - MAM Environment of the NETGEN Final Report
- Appendix B - QMON JOR Listing.

9.2.1 Uses of Disk Queues and Memory Queues

Disk queues are used to store data for later file transfer and processing in batch mode.

Memory queues are used for interactive and transactional dialog when turn-around time for issuing the response is essential.

9.2.2 Structure of a Queued Message

The message is assembled in blocks whose length is determined by the user. Text Control Blocks contain the data submitted on entry and subsequently processed. Header Control Blocks contain the control information required for transmitting data in the TCBs.

The TCB is structured as follows:

- the first 5 bytes provide the link to the associated HCB
- the remaining bytes contain the user data.

There are 3 types of HCBs whose structure depends on their position within the queued message:

1. The first HCB (HCB1) is structured as follows:
   - the first 50 bytes provide the link to HCB2
   - the remaining bytes are structured in multiples of 4 bytes, each multiple providing the link to its TCB.
2. The second HCB (HCB2) is structured as follows:
   - the first 4 bytes provide the link to HCB3
   - the remaining bytes are structured in multiples of 4 bytes, each multiple providing the link to its TCB.

3. the third HCB (HCB3) is structured as follows:
   - the first 4 bytes are set to zero since no subsequent HCB follows
   - the remaining bytes are structured in multiples of 4 bytes, each multiple providing the link to its TCB.

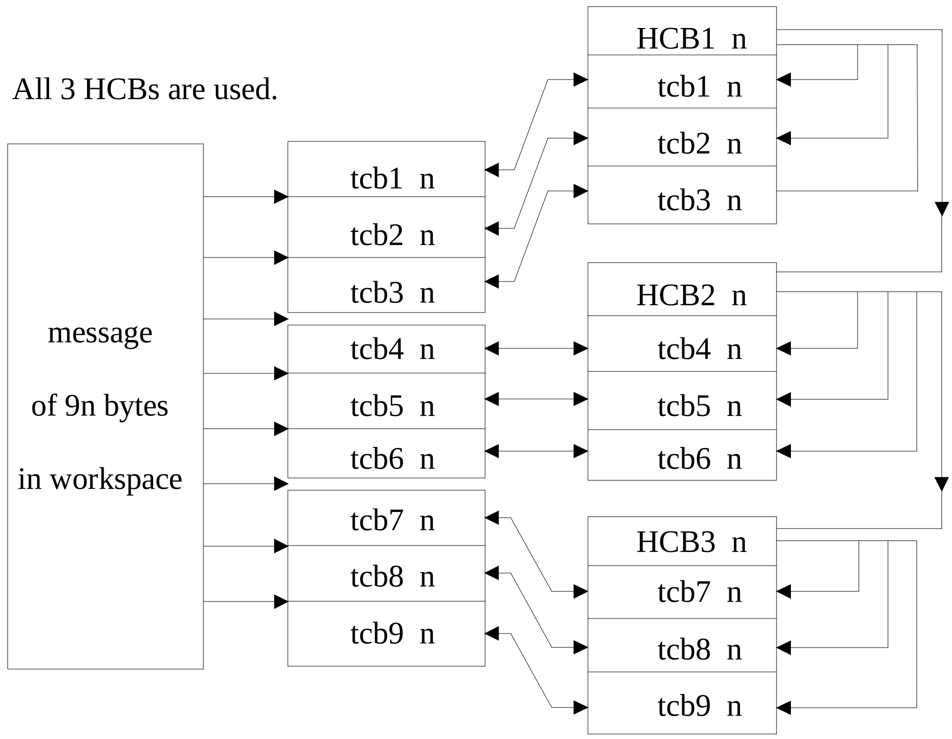
The number of blocks needed to enqueue a message depends on the length of the message. However, the size of the block in bytes is defined in the NETWORK directive by the parameters:

- QFBLKSZ for disk queues
- QMBLKSZ for memory queues.

Case 1:

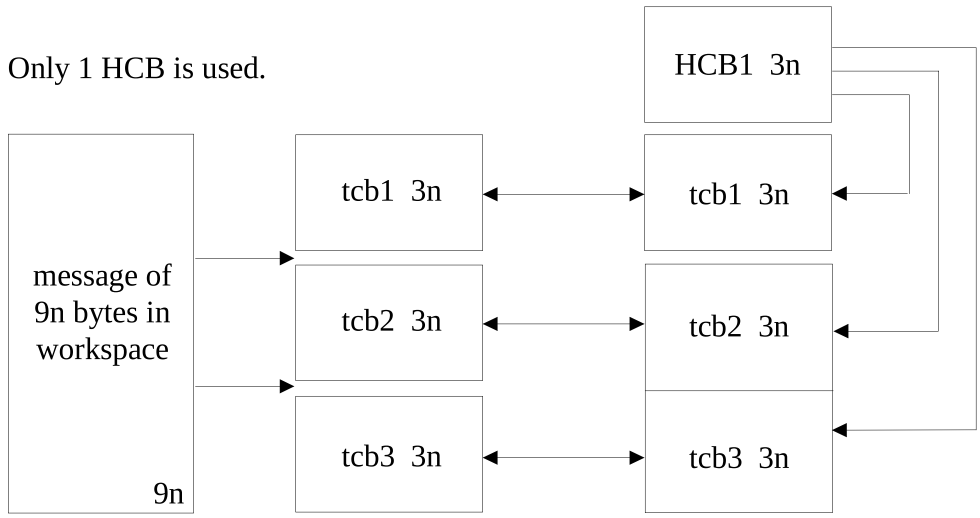
The message is structured using all HCBs (HCB1,HCB2 and HCB3). This means that a message of 9n bytes is structured in multiples of n bytes, giving 9 TCBs. Every 3 TCBs are headed by their appropriate HCB. The number of links required in structuring the message is 11, namely:

- 2 links between the HCBs, HCB1-HCB2 and HCB2-HCB3
- 3 links within each HCB to link all 3 of its TCBs, making a total of 9 for all 3 HCBs.



Case 2:

The message is structured using only 1 HCB. This means that the message of 9n bytes can be structured in multiples, say, of 3n bytes, giving 3 TCBs. The number of links required in structuring the message is 3, namely, the links from the HCB to its TCBs. No other links are involved.



The smaller the block size, the greater the number of links, the longer the time for storage and retrieval, hence the slower the response time. However, the advantage of a smaller block size and hence a greater number of TCBs is that the amount of unoccupied space in the queue is reduced.

9.2.3 Calculating the Number of Blocks

The number of TCBs and HCBs required for structuring the message is given by the following algorithms.

1) The total number of TCBs rounded up to the nearest integer

$$N = \frac{\text{message-length}}{B - 5}$$

where N = number of TCBs
B = block size in bytes

2) Only 1 HCB (HCB1) is used if :

$$4N \leqslant (B - 50)$$

3) 2 HCBs (HCB1 and HCB2) are used if :

$$4N \leqslant (2B - 54)$$

4) All 3 HCBs (HCB1, and HCB2 and HCB3) are used if :

$$4N \leqslant (3B - 58)$$

9.2.4 Specifying QMBLKSZ and QFBLKSZ Values

The block size specified by these parameters is, in the first instance, gauged to be 25% of the average message-length, the proviso being that the gauged size is not less than the defaults provided for at network generation, namely:

- 76 bytes for QMBLKSZ
- 512 bytes for QFBLKSZ.

No value less than the default must be specified. The user can define the block size ranging from the minimum of the default to the maximum of 1024 bytes.

NOTE:

Whatever the size selected, the useful space is always 5 bytes less. The main reason for tuning block size is to ensure rapid turn-around. The criteria on which to gauge block size are:

- to reduce the amount of unoccupied space in the HCB(s) and the last TCB

- to minimize the number of I/Os in the case of a disk queue file.

9.2.4.1 Example of Varying Block Sizes

In this example, a message of 980 bytes has been selected.

The following algorithms have been applied to calculate the values:

1) $\text{block-size} = \dfrac{\text{message-length}}{\text{number of TCBs}} + 5\ \textit{bytes}$ (rounded up)

2) unoccupied space of HCB1 = block-size - (50+(total TCBs * 4) *bytes*

3) unoccupied space last TCB = (block-size - 5) * total TCBs - message_length *bytes*

TCB	Block-Size	Unoccupied Space HCB1	Unoccupied Space last TCB
2	495	437	0
3	332	270	1
4	184	184	0
5	131	131	0
6	95	95	4

The chart shows that as the block size progressively gets smaller, the amount of unoccupied space likewise reduces proportionally.

For this reason, default values for QMBLKSZ and QFBLKSZ are recommended as the initial kick-off sizes and gradually increased until an optimum turn-around is reached.

9.2.4.2 Example of Optimizing the Number of Blocks

Consider a message text to fill a screen of 24 lines, each line containing 80 characters. A block size of 85 bytes has been selected, of which 5 bytes in the block provide the link.

The 2 ways of structuring the message are:

1. Each line is sent as a separate message with SEND WITH EMI.

   Each TCB must therefore have its own HCB1, making 24 TCBs and 24 corresponding HCB1s.

   The total space required is (48 * 85) = 4080 bytes.

   Lines of text are displayed as received without the whole message having to be enqueued. Turn-around time is rapid.

2. Each line is sent as a portion of the message with SENDs and only the last line is sent with SEND WITH EMI.

   The message can be structured with 2 HCBs, namely HCB1 to head the first 8 TCBs and HCB2 to head the remaining 16 TCBs.

   The total space required is (26 * 85) = 2210 bytes.

   The whole message must be enqueued, before transmission starts. However, the saving in space is considerable.

9.2.5 Defining Memory Queue Space

The space in memory for such queues is restricted to a segment of 64 Kbytes.

N = number of memory blocks
= QMPOOL + total MEMBLOCK, where (N * QMBLKSZ) <= 65535 bytes

The term *total MEMBLOCK* is the sum of all the individual MEMBLOCK values declared in the QUEUE directives.

QMPOOL refers to the parameter declared in the NETWORK directive and is available to the queue in which QMPOOL has been specified.

Violating these restrictions causes an error at network generation with the message:

ERROR CN 20 SEVERITY (4) SEGMENT TOO LARGE : segment-name

The actual size of the segment is provided in the NETGEN SYSOUT Report.

Message length, traffic load and the number of sharing queues determine the optimum size of the memory buffer pool.

The tools available to the user for optimizing the buffer pool sizes are :

- the JOR listing at the end of each session
- a hard copy of random snapshots of queue occupancy through DNET PGQ commands issued from time to time during the session.

9.2.6 Defining the Disk Queue File

The disk queue file must be preallocated on a monovolume before it can be used by NETGEN. The size of the disk queue file is a function of:

- QFBLKSZ value in the NETWORK directive determining the block size of the file and proportionally as large as the capacity of the disk itself
- FILEBLOCK value in the QUEUE directive determining the number of blocks for the use of the queues concerned.

  N = max number of blocks in file = total FILEBLOCK <= 32767 byte

```
          13165
T = -------------
     135 + QFBLKSZ
```
, where: T = number of blocks per track

If the total number of blocks exceeds the limits defined by these algorithms, an error is flagged at network generation:

ERROR NG020 SEVERITY (4) SEGMENT TOO LARGE : segment-name

In order to optimize file allocation for disk I/Os and missing segments, the user has the statistics of the JOR listing.

9.3 Allocating Disk Files

Disk files containing user data should be allocated, where possible, on different volumes to reduce contention.

10. Network Configuration Limits

10.1 Basic Network Configuration Limits

10.1.1 Primary Network

Systems		512
Neighbors on PSI and ISL	(n)	108
Passthroughs on PSI		8
Neighbors on ISL		100
Remote Systems		511-n
Session Controls		512
TNS Server		1
FEPS Server Occurrences		8
OCS Server Occurrences		4
OCS controllers		4
OCS Front Ends with IPS configured		4
Sessions Routes		2000
Transport Station on ISL		100
ISL Cables		4
Cable Transceivers (PLs) per System		4
Communication paths outgoing from local LPLs	(c)	4
Network Routes between Local and Remote system	(c*c)	16

10.1.2 Network Administration

Administrative Functions			3
Administrative Groups	(1)		30
Administrative Correspondents	(1)	(2)	420
Administrative Filters			100
Administrative Logfile sets			2
Concurrent Commands from Heterogeneous Systems			255
Responses waiting to be sent to Heterogeneous Systems			255
Commands waiting to be sent to Heterogeneous Systems			255
Messages waiting to be sent to Homogeneous Systems			255
Commands waiting responses from Heterogeneous Systems			255
ACs per AG Concurrent Sessions with an AF			420
Statistics Blocks			500

(1) Includes defaults
(2) Statically configured or dynamically created.

10.1.3 Queue Handling

Declared Queues	255
Disk Queues	255
Memory Queues	255
Memory Pool Blocks	

10.2 Directory Configuration Limits

10.2.1 Correspondents

Declared Objects	10000
Correspondents (XCP1 + XCP2 + TM)	10000

10.2.2 Remote Systems

Remote Systems	5000
Remote Session Controls	5000
ISO PID TSELs	5000
ISO NSAPs	10000
Subdomains	255

10.3 OTHER GENERAL LIMITS

Error of severity 3	50
Number of sessions (VCAM)	5000

❑

A. NETGEN Reports

A.1 NETGEN Sysout Report

A.1.1 NETGEN Header Banner

```
*******************************************************************
*******************************************************************
**** GCOS7                                                     ****
****                    E X E C _ P G                          ****
****                       VERSION: V6.20 DATED: DEC15, 1991   ****
******<NETGEN>************|***********************************|****
*****|********|***********|***********************************|****
     |        |           |                                   |
     +<-Name->+           +<-    version identification    ->+
```

A.1.2 NETGEN Input Active Options

The following examples show how the CRNETGEN command startup parameters are reported.

EXAMPLE 1

```
CRNG CRNET=QQQQ;

*****************************************************************
*        CREATE NETWORK GENERATION : ---> ACTIVE OPTIONS        *
*****************************************************************

DO NOT GENERATE TERMINAL TABLES                      <--- CRTERM
DO NOT GENERATE DIRECTORY (CORR,SYS)                 <--- CRDIR
GENERATE NETWORK COMMUNICATION TABLES                <--- CRNET
  -    USING SOURCE : SYS.DSACONF..QQQQ              <--- SLLIB
  -    DO NOT ENABLE TABLES                          <--- ENABLE
DO NOT SAVE CURRENT GENERATION                       <--- SAVE
DEBUGGING OPTIONS :   -> TEMP          <--- no save and no enable
ADDITIONAL OPTIONS : NONE
```

EXAMPLE 2

```
CRNG CRNET=T_HNG_BP50 CRTERM=STD SLLIB=DSAC.TEST.NG
     SAVE=SYS.DSACORE..HNGBP50 ;

*****************************************************************
*        CREATE NETWORK GENERATION : ---> ACTIVE OPTIONS        *
*****************************************************************

GENERATE STANDARD TERMINAL TABLE                    <--- CRTERM
DO NOT GENERATE DIRECTORY (CORR,SYS)                <--- CRDIR
GENERATE NETWORK COMMUNICATION TABLES               <--- CRNET
  -    USING SOURCE : DSAC.TEST.NG..T_HNG_BP50      <--- SLLIB
  -    DO NOT ENABLE TABLES                         <--- ENABLE
SAVE CURRENT GENERATION AS PRIVATE INTO             <--- SAVE
  -    SYS.DSACORE..HNGBP50                         <--- SAVE
DEBUGGING OPTIONS : NONE
ADDITIONAL OPTIONS : NONE
```

EXAMPLE 3

```
CRNG  CRNET=T_BP50_NET  CRDIR T_BP50_DIR  CRTERM=USERTERM
    SLLIB=DSAC.TEST.NG   SAVE=SYS;

*****************************************************************
*        CREATE NETWORK GENERATION : ---> ACTIVE OPTIONS        *
*****************************************************************

                                                     <--- CRTERM
GENERATE STD TERMINAL TABLE WITH ADDITIONAL COMMANDS TAKEN FROM:
  -    DSAC.TEST.NG..USERTERM                        <--- CRTERM
GENERATE DIRECTORY (CORR,SYS)                        <--- CRDIR
GENERATE NETWORK COMMUNICATION TABLES                <--- CRNET
  -    USING SOURCE : DSAC.TEST.NG..T_BP50_NET       <--- SLLIB
  -    USING SOURCE : DSAC.TEST.NG..T_BP50_DIR       <--- CRDIR
  -    DO NOT ENABLE TABLES                          <--- ENABLE
                                                     <--- SAVE
SAVE CURRENT GENERATION AS SYSTEM INTO SYS.DSACORE..NETGEN
DEBUGGING OPTIONS : NONE
ADDITIONAL OPTIONS : NONE
```

EXAMPLE 4

```
CRNG  CRNET=T_BP50_NET  CRDIR=T_BP50_DIR  SLLIB=DSAC.TEST.NG
    SAVE=SYS  ENABLE;

*****************************************************************
*        CREATE NETWORK GENERATION : ---> ACTIVE OPTIONS        *
*****************************************************************

                                                     <--- CRTERM
DO NOT GENERATE TERMINAL TABLE EXCEPT IF MISSING IN PERM. WORKSP.
GENERATE DIRECTORY (CORR,SYS)                        <--- CRDIR
GENERATE NETWORK COMMUNICATION TABLES                <--- CRNET
  -    USING SOURCE : DSAC.TEST.NG..T_BP50_NET       <--- SLLIB
  -    USING SOURCE : DSAC.TEST.NG..T_BP50_DIR       <--- CRDIR
  -    ENABLE TABLES                                 <--- ENABLE
                                                     <--- SAVE
SAVE CURRENT GENERATION AS SYSTEM INTO SYS.DSACORE..NETGEN
DEBUGGING OPTIONS : NONE
ADDITIONAL OPTIONS : NONE
```

EXAMPLE 5

```
CRNG  CRDIR=T_BP50_DIR  SLLIB=DSAC.TEST.NG
      SAVE=SYS.DSACORE..HNGBP50  ENABLE;

*****************************************************************
*        CREATE NETWORK GENERATION : ---> ACTIVE OPTIONS        *
*****************************************************************

DO NOT GENERATE TERMINAL TABLE                       <--- CRTERM
GENERATE DIRECTORY (CORR,SYS) ONLY                   <--- CRDIR
DO NOT GENERATE NETWORK COMMUNICATION TABLES         <--- CRNET
  -    USING SOURCE : DSAC.TEST.NG..T_BP50 DIR       <--- SLLIB
  -    ENABLE TABLES                                 <--- ENABLE
SAVE CURRENT GENERATION AS PRIVATE INTO              <--- SAVE
  -    SYS.DSACORE..HNGBP50
DEBUGGING OPTIONS : NONE
ADDITIONAL OPTIONS : NONE
```

EXAMPLE 6

```
CRNG  LOAD=SYS.DSACORE..HNGBP50  SAVE=SYS  ENABLE;

*****************************************************************
*        CREATE NETWORK GENERATION : ---> ACTIVE OPTIONS        *
*****************************************************************

RESTORE PREVIOUSLY SAVED TABLES TAKEN FROM           <--- LOAD
  -    SYS.DSACORE..HNGBP50
  -    ENABLE TABLES                                 <--- ENABLE
                                                     <--- SAVE
SAVE CURRENT GENERATION AS SYSTEM INTO SYS.DSACORE..NETGEN
DEBUGGING OPTIONS : NONE
ADDITIONAL OPTIONS : NONE
```

EXAMPLE 7

```
CRNG  LOAD=SYS  ENABLE;

*****************************************************************
*        CREATE NETWORK GENERATION : ---> ACTIVE OPTIONS        *
*****************************************************************

                                                     <--- LOAD
RESTORE PREVIOUSLY SAVED TABLES TAKEN FROM SYS.DSACORE..NETGEN
  -    ENABLE TABLES                                 <--- ENABLE
DO NOT SAVE CURRENT GENERATION                       <--- SAVE
DEBUGGING OPTIONS : NONE
ADDITIONAL OPTIONS : NONE
```

❑

A.1.3 MNTERM Header Banner

The MNTERM header banner appears when terminal table generation is requested.

```
*****************************************************************
*****************************************************************
**** GCOS7                                                   ****
****                     E X E C _ P G                       ****
****                     VERSION: 82.00 DATED: DEC 18, 1987 ****
*****MNTERM*************|**********************************|****
****|******|************|**********************************|****
    |      |            |                                  |
    | Name |<- MNTERM  ->|      Version identification      |
```

A.1.4 MNTERM Input Source Listing

The directives located in the SYS.HSLLIB..H_TERM member (see Appendix A) and, when applicable, those specified in the user member through the parameter CRTERM=usermb are listed as follows:

```
C: DEFTERM EXTYPE=AJ832, DSAMODEL=2015, INTYPE=0C, COMFEAT=PAD,
-: SUBTYPES=(KB,KPR,PRT), PROC1=TTY, PROC2=TTY<-R,
-: FEATURES=(ROLL,HTAB), PRESCLASS=2, TEXTTYPE=10, CSETTYPE=6,
-: PSWCLEAR=27D2, PAGECLEAR=0C070707, SYSHEAD=0D25, SYSTRAIL=0D25,
-: INBFSZ=00, LINEMX=158, MESSLG=128, FORMSTYPE=0, GKSTYPE=1,
-: TABCLEAR=2716, TABSET=27F1, PLHEADER=27C3, PLTRAILER=275E,
-: PLMODE=3, BLOCK=0;
    .
    .
    .
C:DEFLIKE EXTYPE=VIP7805, LIKE=VIP7804, DSAMODEL=4002,
  EMULATE=203E;

C:DEFLIKE EXTYPE=VIP7814, LIKE=VIP7804, DSAMODEL=2047,
  EMULATE=203E;

*****MNTERM*********************************M*N*T*E*R*M*********
```

This line is the MNTERM trailer banner which ends the MNTERM output listing.

A.1.5 NETGEN Input Source

The listing of the directives of the basic network or directory configuration descriptions is of the following form. The first line shows the page header and the first column indicates the sequential numbering of directives as generated by NETGEN.

```
                LISTING OF DIRECTIVES AND ERROR MESSAGES
  0
  1  COMM '=====================================================';
  2  COMM '===         V10  NETGEN COMMUNICATIONS          ===';
  3  COMM '=====================================================';
  4
  5  COMM '===               LOCAL  SYSTEM                 ===';
  6
  7  SYS   BP50  PF=LSYS  SCID=70:80;
  8
  9  COMM '===         DESCRIPTION OF FRONTENDS            ===';
 10  COMM '===                        SERVERS              ===';
 11  COMM '===                        PSI LINKUPS          ===';
 12
 13  DEF   CT  SET  WATCH=0;
 14  SYS   BP51, PF=DN7100, PSI=CC28, SCID=70:81, POOLSZ=(2,5);
 15  SYS   BP1E, PF=DN7100, PSI=CC01, SCID=70:30, POOLSZ=(2,5);
     .
     .
     .
391  COMM '=====================================================';
392  COMM '===       DEPARTMENTALIZED DPS 6 STATIONS       ===';
393  COMM '=====================================================';
394
395  SYS   L6AB, PF='DPS6/DSS/2.0', SCID=65:81, PT=(BP51,BP1E);
396  SYS   L6AD, PF='DPS6/DSS/2.0', SCID=65:84, PT=(BP51,BP1E);
397  SYS   L6AE, PF='DPS6/DSS/2.0', SCID=65:85, PT=(BP51,BP1E);
398  SYS   L6AF, PF='DPS6/DSS/2.0', SCID=65:86, PT=(BP51,BP1E);
399  SYS   L6AI, PF='DPS6/DSS/2.0', SCID=65:109,PT=(BP51,BP1E);
400  SYS   L6AK, PF='DPS6/DSS/2.0', SCID=65:91, PT=(BP51,BP1E);
401  SYS   L6AN, PF='DPS6/DSS/2.0', SCID=65:94, PT=(BP51,BP1E);
402  SYS   L6AU, PF='DPS6/DSS/2.0', SCID=65:80, PT=(BP51,BP1E);
403  SYS   L6A1, PF='DPS6/DSS/2.0', SCID=75:01, PT=(BP51,BP1E);
404  SYS   L6A2, PF='DPS6/DSS/2.0', SCID=75:02, PT=(BP51,BP1E);
405  SYS   L6A3, PF='DPS6/DSS/2.0', SCID=75:03, PT=(BP51,BP1E);
406  SYS   L6A4, PF='DPS6/DSS/2.0', SCID=75:04, PT=(BP51,BP1E);
407  SYS   BC28, PF='DPS6/DSS/2.0', SCID=65:40, PT=(BP51,BP1E);
408  SYS   BC29, PF='DPS6/DSS/2.0', SCID=65:41, PT=(BP51,BP1E);
409  SYS   BC2A, PF='DPS6/DSS/2.0', SCID=65:42, PT=(BP51,BP1E);
410  SYS   BC2B, PF='DPS6/DSS/2.0', SCID=65:43, PT=(BP51,BP1E);
420  ENDNETWORK  BP50;
```

A.1.6 NETGEN Error Reporting

Error Message

Errors detected during the syntax analysis phase are flagged following the line in error.

Each message comprises the following parts from left to right:

`***`	As many asterisks as the severity level of the error
`ERROR NGnnn`	where nnn is a 3 digits decimal integer representing the error number which can be used as entry in the paragraph 'NETGEN error message index' (later in this Appendix) for details about each error message
`SEVERITY n`	Where n is a one digit integer ranging from 2 to 4 with the following conventions:
`2`	WARNING level: the error condition is light and does not jeopardise the configuration execution
`3`	FATAL level: the error condition is serious and may abort NETGEN processing; in some cases (syntax analysis phase for instance) the processing goes on to completion so that other possible errors may be detected
`4`	FATAL STRONG level: the error condition imposes the immediate abort of the NETGEN processing
`error message`	it explains the error condition. Sometimes it is splitted into two consecutive parts separated by a colon.The first gives the general explanation,the second specifies the particular object involved
`RC=hexa`	optionally it indicates the internal return code returned by NETGEN.

Errors attached to a single directive are issued just after this directive. Errors involving several lines or the whole source are issued at the end of the source listing.

EXAMPLE of Single Error

```
155    SR    RSC=BP3C, SVR=BP1E, STATE=LOCK;
   .01 *** ERROR      NG003 SEVERITY 03 - DUPLICATED : BP3CBP1E
```
❑

EXAMPLE of Multiple Errors in a Line

```
 51    RSC   SPY, SVR=TNS, PROTOCOL=DSA200, RTS=SPY,
       SCID=70:70, TSAP=(ISODSA:BP46);
   .01 *** ERROR     NG008 SEVERITY 03 - INVALID VALUE : 70
   .02 *** ERROR     NG003 SEVERITY 03 - DUPLICATED : BP46
```
❑

EXAMPLE of Global Errors involving Several Lines

```
**** ERROR NG053 SEVERITY 04-COMMAND <SR>NOT DECLARED:BP3CSVR_BP1E
```
❑

EXAMPLE of Global Errors involving the Whole Description

When CRDIR is used one of the following messages may appear after the ENDDIRECTORY directive:

```
*  NO CORRESPONDENT OBJECT REGISTERED IN THE DIRECTORY.
*  NO REMOTE SYSTEM OBJECT REGISTERED IN THE DIRECTORY.
** NO OBJECT REGISTERED IN THE DIRECTORY.
```
❑

Error Summary

An error summary report is issued after the source listing and all detailed error messages possibly scattered in it.It is of the form:

```
     SEVERITY 2 WARNINGS    :   nnn
     SEVERITY 3 ERRORS      :   0nn
     SEVERITY 4 ERRORS      :   00n
```

NOTES:

1. If several errors in the same directive are detected by the GCL analyzer, all are printed but are computed by the analyzer as a single severity 3 error in the report.
2. By definition there can be only one severity 4 error. The number of severity 3 is limited to 50, beyond this count a severity 4 error condition is raised. The total of warnings is limited to three decimal digits.
3. The error summary does not take into account the errors occuring after the generating step itself, that is, errors during the saving or enabling steps.

If generation processing is error-free, the error summary is replaced by the following message:

```
NO ERROR DETECTED
```

Depending on the affect on the running communications subsystem, if there is one, the error reporting ends always by one of the two following messages:

```
ENABLED NG (IF ANY) PRESERVED
ENABLED NG (IF ANY) LOST
```

A.1.7 NETGEN Final Report

This report is issued by NETGEN after error-free completion of all the processing requested by the input parameters.

```
*************************NETWORK ENVIRONMENT********************

GENERATION INFORMATION                                  <--- always
   CONFIGURATION NAME         = name                    <--- from NET
   LOCAL SYSTEM NAME          = name                    <--- from LSYS
   PERMANENT WORKSPACE NAME   = CTMN                    <--- for basic
   BASIC GENERATION STAMP     = date     time           <--- if CRNET
   DIRECTORY GENERATION STAMP = date     time           <--- if CRDIR

SAVE-LOAD INFORMATION :                           <--- if SAVE or LOAD
   CREATION TIME RELEASE = V600                          <--- at least
   CREATION TIME STAMP   = .UG18 9 0 / BP50 /X 42 <--- by example

                                                     *************

BASIC NETWORK CONFIGURATION TABLES:    <--- always  *  N. U. V. *
                                                    * (segment) *
   ROOT TABLE      =   nnnnn BYTES                   9.23 | D.46
   GCB TABLE       =   nnnnn BYTES                   9.22 | D.45
   TNS TABLE 1     =   nnnnn BYTES                   9.25 | D.50
   TNS TABLE 2     =   nnnnn BYTES                   9.xx | 9.50
   TNS TABLE 3     =   nnnnn BYTES                   9.xx | 9.51
   QMCS TABLE      =   nnnnn BYTES                   9.21 | D.42
   MAM QM POOL     =   nnnnn BYTES                   9.20 | D.41
   MAM QF POOL     =   nnnnn BYTES                   9.17 | D.40
   VCAM TABLE 1    =   nnnnn BYTES                   9.1D | D.3E
   VCAM TABLE 2    =   nnnnn BYTES                   9.27 | D.52
   VCAM TABLE 3    =   nnnnn BYTES                   9.1E | D.3F
   VCAM BUF POOL   =   nnnnn BYTES                   9.24 | D.4A
   FEPS TABLE      =   nnnnn BYTES                   9.28 | D.53
   SEMAPH. TABLE   =   nnnnn BYTES                   9.29 | D.55
   DICTIONARY      =   nnnnn BYTES                   9.26 | D.51
   RAEH TABLE 1    =   nnnnn BYTES                   9.xx | A.xx
   RAEH TABLE 2    =   nnnnn BYTES                   9.xx | F.5F
   RAEH TABLE 3    =   nnnnn BYTES                   9.xx | B.E8
                                                    * CUR | ENBL*
                                                    *************
```

```
TUNING INFORMATION                                         <--- if CRNET
   TOTAL SYSTEMS                =    nnn
                 WHICH OCCUPY  -> nnnnnn BYTES
   TOTAL DSA SESSIONS CONTROLS  =    nnn
                 WHICH OCCUPY  ->  nnnnn BYTES
   TOTAL BASIC SESSIONS ROUTES  =   nnnn
                 WHICH OCCUPY  ->  nnnnn BYTES
   TOTAL TRANSPORT STATIONS     =     nn
                 WHICH OCCUPY  ->  nnnnn BYTES
   TOTAL SERVERS                =     nn
                 WHICH OCCUPY  ->  nnnnn BYTES
   TOTAL ISL NETWORK ROUTES     =    nnn
                 WHICH OCCUPY  ->  nnnnn BYTES
   TOTAL ISL REMOTE ATTACHMENTS =    nnn
                 WHICH OCCUPY  ->  nnnnn BYTES

                                          <--- if queues declared
********************************** MAM ENVIRONMENT **************

DECLARED MEMORY QUEUES  =      nn          <--- MBLK + sum QMPOOL

MEMORY QUEUE POOL       =   nnnnn BYTES    <--- ((QMPOOL+sumMBLK)
                                                         * QMBLKZ)

DECLARED DISK QUEUE(S)  =      nn                  <--- from FBLK

FILE QUEUE POOL         =   nnnnn BYTES    <-- computed by NETGEN

QUEUE FILE INFO :                                   <--- if QFILE

        EFN    = qfilename
        MEDIA  = media
        SIZE   = nnnnn REC.

MAM TABLES SIZE  =  nnnnn BYTES                     <--- for  MAM

Q.M. TABLES SIZE = nnnnn BYTES                       <--- for QMON

                                                      <--- always

*********************** ADMINISTRATIVE ENV. *********************

   TOTAL STATISTICS BLOCKS               =     nnn
                           WHICH OCCUPY ->  nnnnnn  BYTES

                                                      <--- always
```

```
******************************** DIRECTORY TCORR-TSYS************

CORRESPONDENT DICTIONARY (TCORR) IS EMPTY.          <--- if empty
CORRESPONDENT DICTIONARY (TCORR) CONTENTS :     <--- if not empty
   TOTAL SPACE ACTUALLY USED            = nnnnnnn  BYTES
   TOTAL SPACE PERCENTAGE USED          =      nn  %
    -> TOTAL CORRESPONDENT NUMBER       =   nnnnn  (nn %)
    -> (R F U)                          =          (   %)
    -> TOTAL XCP2 POOL NUMBER           =     nnn  (nn %)
    -> TOTAL WORKSTATION NUMBER         =     nnn  (nn %)

REMOTE SYSTEM DICTIONARY (TSYS) IS EMPTY.           <--- if empty
REMOTE SYSTEM DICTIONARY (TSYS) CONTENTS :      <--- if not empty
   TOTAL SPACE ACTUALLY USED            = nnnnnnn  BYTES
   TOTAL SPACE PERCENTAGE USED          =      nn  %
    -> TOTAL DIRECTORY REMOTE SYSTEMS   =    nnnn  (nn %)
    -> TOTAL DIRECTORY SESSION CONTROLS =    nnnn  (nn %)
    -> TOTAL DIRECTORY ISO SUB-DOMAINS  =     nnn  (nn %)
    -> TOTAL LOCAL SYSTEM ISO ADDRESS   =       n  ( n %)
    -> TOTAL BASIC NETWORK SYSTEMS      =     nnn  (nn %)
    -> TOTAL BASIC NETWORK SESSIONS     =     nnn  (nn %)
    -> TOTAL BASIC NETWORK PT SESSIONS  =      nn  (nn %)
    -> TOTAL BASIC NETWORK PT SERVERS   =      nn  (nn %)
    -> TOTAL BASIC NETWORK PT REMOTE TS =      nn  (nn %)
    -> TOTAL SYSTEM PROFILES            =      nn  (nn %)
    -> TOTAL GCL ENVIRONMENT OBJECTS    =     nnn  (nn %)
    -> TOTAL CONFIGURED ISO NSAP'S      =    nnnn  (nn %)

                                        <--- end of report banner
*****************************************************************
```

NOTES:

1. Percentages are generally expressed by the nearest integer value except for 0 which stands for *actually zero* instead of *less than 0.5*.
2. Percentages on lines headed by -> are partial percentages computed from TOTAL SPACE ACTUALLY USED written above them.
3. The amount of space available in each directory can be evaluated from the TOTAL SPACE PERCENTAGE USED.
4. The SAVE - LOAD stamp must be seen as a *unique occurrence string* attached to each of the three configuration members and internally used for coherency checking purpose. It is reported only for visual awareness.

A.2 NETGEN Error Messages

The error messages output from the execution of the NETGEN utility, appear in the SYSOUT file.

Where an abnormal condition occurs on writing to the SYSOUT file, a specific error message is issued to the JOR.

There are two main kinds of errors reported on the SYSOUT file by NETGEN processing:

- syntactical errors issued directly by the GCL command analyzer
- errors detected by NETGEN during semantic checkings or in the processing phases launched after directive analysis.

A.2.1 GCL Analyzer Error Message Format

The general layout of such error reporting is of the form:

- the erroneous line is printed again, right shifted and headed by the prompt: ?:
- followed by a line containing as many error flags "?" as error conditions detected on the erroneous line. Each flag pointing upwards a term in error
- following lines are a blank line and as many error message lines as error flags; each error message is headed by "***" standing for severity 3 level (fatal errors).

The consequences of such errors are:

- after at least one error of this kind, the NETGEN will end just after the last command of the source has been analyzed
- all these errors are totalized in the severity 3 error count of the final error report, with the exception that each multi-error line is counted as only one error
- the corrective action is always the same: correct the error cause identified by the error message and launch NETGEN utility again.

Since all these errors are of the same class (syntax error), their cause is generally well identified by the message and the action for correcting them easy to infer, so:

- no numbering is attached to them
- they are not detailed in the *NETGEN Error Message Index* in this Appendix.

A.2.2 NETGEN Error Message Format

The general outline of a NETGEN error message is as follows:

`**** ERROR NGnnn SEVERITY (s) error-message [RC=rc], where:`

****	As many stars as indicated by the severity level,
NG	denotes that the source of the message is the NETGEN utility,
nnnn	indicates the message number
s	indicates the degree of severity of the error:
	2 Warning, the network generated may still be usable
	3 Fatal, leads to an abort of NETGEN execution but triggers a complete syntax analysis of the network
	4 Fatal strong, leads to an abort of NETGEN execution and may also stop syntax analysis.
error-message	explains the error condition
RC	optionally the error message is followed by the internal return code returned by NETGEN under the following form:
	RC=hexa8: the actual contents of the RC register
	->siu ic,rc: gives the name and component number (ic) of the system integration unit (siu) which has set the current return code followed by the character representation (rc) of the return code itself.

A.3 NETGEN Error Message Index

NETGEN error message detailed in this index are ordered in their increasing error number sequence.

In this index the error line of each entry is shown as actually printed in the report except that:

- the heading stars and the first word ERROR are suppressed
- the variable parts in the message are represented in lower case and explained in the *Syntax* paragraph which follows them
- the string RC=rc stands for the return code outline as precedingly explained.

A.3.1 JOR Messages

In the JOR the format is simplified as follows:

```
NGnnn     error-message.
```

```
NG002     ERROR DURING OPEN  SYSOUT RC=rc
NG002     ERROR DURING PUT   SYSOUT RC=rc
NG002     ERROR DURING CLOSE SYSOUT RC=rc
```

Cause: RC specifies the error and the system primitive. NETGEN execution has been aborted due to an error while accessing the SYSOUT file as indicated in the message.

Action: Retry.

A.3.2 SYSOUT Error Messages

NG003 SEVERITY 3 **DUPLICATED:** *name*

Cause: *name* identifies the object in error. Names for a given object type must be network-wide and unique.

Action: Use another name in the appropriate directive and rerun NETGEN.

NG006 **SEVERITY 3 DIRECTORY NAME DOES NOT MATCH THAT OF BASIC CONFIGURATION**

Cause: For an incremental generation the name declared in DIRECTORY directive does not match that declared in NETWORK directive.

Action: Modify the input source description and rerun NETGEN.

NG007 SEVERITY 3 **USELESS KEYWORD IN NETWORK:** *name*

Cause: *name* identifies the erroneous keyword QMPOOL or MAMSIMU or PASSWORD of the NETWORK directive:

- either keywords MAMSIMU or PASSWORD have been specified in the NETWORK directive and no QUEUE directive is declared
- or QMPOOL is specified in the NETWORK directive and no memory queue have been declared.

This may lead to a waste of memory space, such as the generation of useless tables. These errors are also detected by the GCL analyzer when the ENDNETWORK directive is present.

Action: Remove the erroneous parameter from the NET directive or add the relevant QUEUE directives and rerun NETGEN.

NG009 SEVERITY 3 **SESSION CONTROL NOT DECLARED :** *name*

Cause: *name* identifies the session control in error. This session has not been found in any RSC directive of the basic network or directory configuration descriptions.In case of incremental generation the basic network configuration involved is that currently running.

Action: Add or modify the RSC directive in the appropriate configuration and rerun NETGEN.

NG010 SEVERITY 3 **CORRESPONDENT NOT DECLARED :** *name*

Cause: *name* identifies the correspondent referenced either in a LGAPPL directive or in a POOL directive. This correspondent has not been found in any TMCOR, XCP1COR or XCP2COR directive.

Action: Add the relevant directive and rerun NETGEN.

NG011 SEVERITY 3 **WORKSTATION NOT DECLARED :** *name*

Cause: *name* identifies a TDS workstation referenced in an XCP1COR directive or an XCP2 workstation referenced in a POOL, TDSWKS or XCP2COR directive. This workstation has not been found in any TDSWKS or XCP2WKS directive.

Action: Add or modify the relevant directive and rerun NETGEN.

NG013 SEVERITY 4 **UNABLE TO CREATE:** *name* RC=rc

Cause: *name* may be that of a workstation, a mailbox, a segment table (internal name) or an object to be put in dictionary.
The contents of RC defines the error and the system primitive.
A system malfunction has prevented the NETGEN utility from creating or extending the specified object.

Action: Call the Service Center.

NG014 SEVERITY 4 **NO SOURCE EXPECTED FOR DIRECTORY**

Cause: The CRDIR parameter has not been specified in CRNETGEN command but at least one directive specific of the directory configuration has been encountered.

Action: Check the consistency between the options of CRNETGEN command and the specified input source descriptions and rerun NETGEN.

NG014 SEVERITY 4 **NO SOURCE EXPECTED FOR NETWORK**

Cause: The CRNET parameter has not been specified in CRNETGEN command but at least one directive specific of the basic network configuration has been encountered.

Action: Check consistency between the options of CRNETGEN command and the specified input source descriptions and rerun NETGEN.

NG015 SEVERITY **ALLOWED MAXIMUM NUMBER OF CONFIGURABLE SYSTEMS IS EXCEEDED**

Cause: The maximum number of systems configurable in the basic network configuration is exceeded. See limits in Section 10.

Action: Move some or all of the remote systems into the directory configuration.

NG015 SEVERITY **ALLOWED MAXIMUM NUMBER OF CONFIGURABLE SR IS EXCEEDED**

Cause: The maximum number of session routes statically configurable within the basic network configuration is exceeded. See limits in Section 10.

Action: Move some or all of the remote systems into the directory configuration, to establish the session routes dynamically.

NG016 SEVERITY 4 **TOO MANY ERRORS : PROCESSING ABORTED**

Cause: More than 50 errors with fatal severity level (3) have been encountered during analysis.

Action: Correct the already reported errors and rerun NETGEN.

NG017 SEVERITY **NO RPL HAS BEEN ATTACHED TO :** *name*

Cause: *name* identifies the cable declared by the CB parameter of the RPL directive and must have all the mandatory parameters (NAME, RCT and IADDR).

Action: Correct the RPL directive and rerun NETGEN.

NG018 SEVERITY **NOT FOUND IN DICTIONARY:** *name*

Cause: *name* identifies the object in error. The object requested for linking should already be registered in the basic network dictionary at the time of search.

Action: Add the appropriate directive and rerun NETGEN.

NG019 SEVERITY **UNABLE TO INITIALIZE DIRECTORY TCORR-TSYS** RC=rc

Cause: RC specifies the error and the system primitive. An anomaly has occurred when initializing the directory.

Action: Call the Service Center.

NG020 SEVERITY 4 **SEGMENT TOO LARGE:** *name*

Cause: *name* identifies a segment table (by its internal name) or a memory queue. For memory queue the value(s) declared for MEMBLOCK in the QD directive multiplied by that declared for QMBLKSZ in the NETWORK directive exceeds 65535. Otherwise the segment table exceeds 65535.

Action: Adjust the value(s) for the NETWORK and/or QD directive, as appropriate, and rerun NETGEN.

NG021 SEVERITY 4 **UNABLE TO OPEN INPUT FILE** RC=rc

NG021 SEVERITY 4 **UNABLE TO READ INPUT FILE** RC=rc

NG021 SEVERITY 4 **UNABLE TO CLOSE INPUT FILE** RC=rc

Cause: RC specifies the error and the system primitive. User or system error.

Action: Check the GCL statement for user errors and rerun NETGEN. If this same error recurs, call the Service Center.

NG022 SEVERITY 4 **EXPECTED SOURCE MISSING FOR DIRECTORY**

Cause: The CRDIR parameter has been specified in the CRNETGEN command but no directive related to the directory configuration description has been encountered before the end of the file.

Action: Check for consistency between the options of CRNETGEN command and the contents of the specified input configuration descriptions and rerun NETGEN.

NG022 SEVERITY 4 **EXPECTED SOURCE MISSING FOR NETWORK**

Cause: The CRNET parameter has been specified in the CRNETGEN command but no directive related to the basic network configuration description has been encountered before either the DIRECTORY directive or the end of the file.

Action: Check for consistency between the options of CRNETGEN command and the contents of the specified input configuration descriptions and rerun NETGEN.

NG023 SEVERITY 4 **QUEUES FILE ACCESS ERROR** RC=rc

Cause: RC specifies the error as well as the system primitive involved. The error, when accessing queue files, may be:
- either a user error, when the file has not been preallocated
- or a system error, if IOFAIL.

Action: - If the file has not been preallocated, preallocate it using the BUILD_MCS_FILE GCL command.
- If "IOFAIL" return code has occurred, rerun NETGEN. If this same condition recurs, call the Service Center.

NG024 SEVERITY 4 **NON STANDARD DISK**

Cause: The disk queue file has not been preallocated on a disk media, the device class of which must start with MS/.

Action: Deallocate the file and reallocate it on an appropriate media and rerun NETGEN.

NG025 SEVERITY 4 **QUEUES FILE TOO LARGE**

Cause: The size of the preallocated disk queue file is in excess of the limit allowed.

Action: Adjust the file size in the appropriate QUEUE directive and rerun NETGEN.

NG026 SEVERITY **PARAMETERS CONFLICTS: ROLLBACK/MEM QUEUE**

Cause: The RESTART parameter of the QUEUE directive has been specified for memory queues declared for the network, by:
- either the QMBLKSZ and QMPOOL parameter of the NETWORK directive
- or the MEMBLOCK parameter of the QUEUE directive.

Action: Delete the erroneous parameter from the concerned directive, and rerun the NETGEN utility.

NG026 SEVERITY **PARAMETERS CONFLICTS: AG INIT=AC/AC BACKUP.**

Cause: AC BACKUP is only allowed when the AG which the AC belongs to is declared with INIT=AF.

Action: Either correct the AG directive by adding INIT = AC or suppress BACKUP in the AC directive and rerun NETGEN.

NG026 SEVERITY **PARAMETERS CONFLICTS: AG INIT=AF/SEVERAL PRIMARY AC.**

Cause: Only one primary AC is allowed under an AG with INIT = AF; all the others must have BACKUP.

Action: Add BACKUP in the appropriate AC's and rerun NETGEN.

NG026 SEVERITY **PARAMETERS CONFLICTS: (SC, MBX) ALREADY DECLARED.**

Cause: The couple of names SESS, MBX appearing in AC directives must appear only once.

Action: Correct appropriately the AC directive and rerun NETGEN.

NG027 SEVERITY 4 **ERROR DURING GCL INITIALIZATION**

Cause: An abnormal condition occurred during the opening of the NG domain.

Action: Check for the NG domain being in the SYS.HBINLIB or call the Field Service Center.

NG027 SEVERITY **ERROR DURING GCL READ:** *reason*

Cause: When present the additional *reason* is intended to help field engineering personnel. Abnormal condition occurred within GCL directive analyzer.

Action: Call the Service Center.

NG027 SEVERITY 4 **ERROR DURING GCL TERMINATION**

Cause: An abnormal condition has occurred while closing the GCL analyzer session.

Action: Call the Service Center.

NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB-> NETWORK** RC=rc
NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB-> TERMINAL** RC=rc
NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB-> DIRECTORY** RC=rc

Cause: RC specifies the error and the system primitive. An abnormal condition has occurred while retrieving some part of the configuration from the specified binlib.

Action: Check the involved binary library contents, correct the CRNETGEN command and rerun the NETGEN utility.

NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB TEMPORARY NETWORK**
NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB " TERMINAL**
NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB " DIRECTORY**

Cause: Specified part of the configuration retrieved in the specified binary library has been saved while in temporary mode (TEMP=1 in CRNETGEN command), so it cannot be handled by the loading function.

Action: Check the involved binary library contents, correct the CRNETGEN command and rerun the NETGEN utility.

NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB->OBSOLETE NETWORK**
NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB->" TERMINAL**
NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB->" DIRECTORY**

Cause: Specified part of the configuration retrieved in the specified binary library has been saved by an obsolete version of NETGEN running under previous GCOS 7 releases, resulting in a configuration incompatible with release V6. This configuration cannot subsequently be handled by the loading function.

Action: Check the involved binary library contents, correct the CRNETGEN command as appropriate and rerun NETGEN.

NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB->ADVANCED NETWORK**
NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB->" TERMINAL**
NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB->" DIRECTORY**

Cause: Specified part of the configuration retrieved in the specified binary library has been saved by an advanced version of NETGEN running under future GCOS 7 release resulting in a configuration incompatible with release V6. This configuration cannot subsequently be handled by the loading function.

Action: Check the involved binary library contents, correct the CRNETGEN command as appropriate and rerun NETGEN.

NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB ->**
{**DIRECTORY | TERMINAL**} **NOT COHERENT WITH BASIC**

Cause: Specified part of the configuration retrieved in the binary library has not been saved by the same run occurrence of NETGEN as for the basic network configuration, resulting in an inconsistent configuration which cannot subsequently be handled by the loading function.

Action: Check the involved binary library contents, correct the NETGEN directive and rerun the NETGEN utility.

NG030 SEVERITY 3 **UNABLE TO LOAD FROM BINLIB -> INTERNAL ERROR**
RC=rc

Cause: RC specifies the system error and the system primitive.

Action: Call the Service Center.

NG031 SEVERITY 3 **UNABLE TO ACCESS WORKSPACE** RC=rc

Cause: RC specifies the error and the system primitive. An abnormal system condition has occurred while trying to access the permanent workspace containing the configuration loaded from binlib.

Action: Call the Service Center.

NG033 SEVERITY 4 **UNABLE TO STORE NEW TERMINAL TABLE**

Cause: Abnormal system error while trying to store the newly created terminal table into temporary or permanent workspace.

Action: Call the Service Center.

NG034 SEVERITY 3 **UNABLE TO SAVE INTO BINLIB NETWORK** RC=rc
NG034 SEVERITY 3 **UNABLE TO SAVE INTO BINLIB TERMINAL** RC=rc
NG034 SEVERITY 3 **UNABLE TO SAVE INTO BINLIB DIRECTORY** RC=rc

Cause: RC specifies the error and the system primitive. A user or system error has occurred while trying to save some part of a generated or loaded configuration from temporary or permanent workspace into specified binary library.

Action: Check the return code. If possible, correct and rerun NETGEN; otherwise call the Service Center.

NG034 SEVERITY 4 **UNABLE TO SAVE INTO BINLIB -> INTERNAL ERROR** RC=rc

Cause: RC specifies the error and the system primitive. A system error.

Action: Call the Service Center.

NG035 SEVERITY **NETGEN CANNOT RUN UNDER THE CURRENT GCOS7 VERSION**

Cause: Attempt to start NETGEN V6 under an previous GCOS-7 release not supporting V6 communications.

Action: Call Service Center for GCOS-7 version switching.

NG036 SEVERITY 4 **QFILE SPECIFIED BUT NO FILE QUEUES DECLARED**

Cause: A sequential file has been declared and assigned but no file queue occurred in the network description.

Action: Either suppress the file declaration or add the missing QD directives and rerun NETGEN.

NG037 SEVERITY **QFILE EFN DOES NOT MATCH THAT CONTAINED IN SAVED GENERATION.**

Cause: The disk queue file currently assigned has not the same efn as the file actually declared at time of generation and save of the configuration presently being restored and loaded.

Action: Assign a file with same efn as at generation time and rerun NETGEN.

NG038 SEVERITY **UNABLE TO STORE DIRECTORY TCORR-TSYS** RC=rc

Cause: RC specifies the error and the system primitive. An error has occurred while storing the generated directory configuration into the temporary or permanent workspace.

Action:
- if INVUSE (returned by VMM system component), reinitialize the GCOS 7 with option CLEAN
- if NOMATCH, call Service Center
- for other values, try rerunning NETGEN.

NG039 SEVERITY 3 **UNABLE TO ENABLE DIRECTORY TCORR-TSYS reason**
RC=rc

Cause: RC specifies the error and the system primitive. An error has occurred while enabling the directory configuration. The reason stated in the message specifies whether if the basic configuration is altered or not by this abort.

Action: For incremental generation two cases may occur:
- The basic network configuration is altered: No valid directory is installed and no further incremental enabling is possible. The current telecommunication session works in a lowered mode: the network operator has to request promptly a weak shutdown and to rerun NETGEN utility to settle again the basic network configuration.
- The basic network configuration is not altered: The preceding directory is disabled but incremental enablings may be attempted. If unsuccessful, call the service center.
For the basic network and directory generation, the associated basic network configuration is then declared not valid. Then retry another basic network generation with or without directory or call the Service Center.

NG041 SEVERITY 3 **MORE THAN 255 QUEUE COMMANDS**

Cause: The number of QD directives declared in the network generation has exceeded the implementation limit.

Action: Reduce the number of corresponding objects to or under the specified limit and rerun NETGEN.

NG041 SEVERITY 3 **MORE THAN 30 AGs DECLARED**

Cause: The number of AG directives declared in the network generation has exceeded the implementation limit.

Action: Reduce the number of corresponding objects to or under the specified limit and rerun NETGEN.

NG041 SEVERITY 3 **MORE THAN 420 ACs DECLARED**

Cause: The number of AC directives declared in the network generation has exceeded the implementation limit.

Action: Reduce the number of corresponding objects to or under the specified limit and rerun NETGEN.

NG042 SEVERITY 4 **UNABLE TO CREATE OR RETRIEVE:** *name*

Cause: *name* identifies the object concerned by the error. Either the object cannot be created into a communications dictionary or it cannot be retrieved in it. This error can also be caused if more than 8 objects have the same name.

Action: Check the description source for this object and if no error found call the Service Center.

NG044 SEVERITY **BACKUP NOT DECLARED :** *name*

Cause: *name* identifies the backup correspondent referenced in an TMCOR, XCP1COR or XCP2COR directive. This backup correspondent has not been found in any TMCOR, XCP1COR or XCP2COR directive.

Action: Add or modify the relevant directive and rerun NETGEN.

NG050 SEVERITY 4 **UNABLE TO CREATE NETWORK FILE** RC=rc

Cause: RC defines the error and the system primitive. NETGEN has aborted, due to an abnormal condition occurring during the copying of communications tables into the temporary or permanent workspace.

Action Retry. If repeatedly unsuccessful perform ISL with the RESTART CLEAN option and rerun NETGEN.

NG051 SEVERITY 4 **UNABLE TO EXPAND A SEGMENT SIZE** *pointer size* RC=rc

Cause: RC defines the system error and the primitive. When present, *pointer* and *size* are intended for more detailed diagnosis. A system error.

Action: Call the Service Center.

NG052 SEVERITY 4 **SAME NETWORK ROUTE MAPPED ON SEVERAL RTS:** *name*

Cause: *name* identifies the network route in error. Incorrect network definition. Any given network route may be allocated to only one Remote Transport Station.

Action: Correct the configuration description accordingly, and rerun NETGEN.

NG053 SEVERITY 3 **LOW LEVEL DIRECTIVE NOT DECLARED:** *xx name*

Cause: *xx* identifies the missing low level directive.
name identifies the referenced object.
A low-level directive references an object which must be defined by another low-level directive which is absent.

Action: Correct the configuration description by inserting the missing low level directive, and rerun NETGEN.

NG054 SEVERITY 4 **NETWORK CONFIGURATION OVERFLOW**

Cause: Some NETGEN limits have been exceeded.

Action: Correct the configuration description with special regard to configurability limits in Section 10, then rerun NETGEN.

NG055 SEVERITY 3 **LT LSAP=ISO IS MANDATORY FOR TNS**

Cause: When LT LSAP=TCPIP is decared without LT LSAP=ISO, the TNS server cannot be used by the ISO transport.

Action: Correct the configuration description by declaring another LT LSAP=ISO, so that the TNS server can be used by the ISO transport, then rerun NETGEN. Alternatively, do not make the declaration but remove SVR=TNS from the RSC directive.

NG056 SEVERITY **LOW LEVEL DIRECTIVE NOT REFERENCED:** *xx name*Cause:
xx identifies the redundant low level directive.
name identifies the referenced object.
An object has been defined but not referenced by any other low-level directive.

Action: Check that the low-level directive in question is not an error or add the missing low-level directives to the configuration or suppress the redundant one and rerun NETGEN.

NG057 SEVERITY 4 **ERROR DURING** {**OPEN | PUT | GET | CLOSE**} **WORK FILE** RC=rc

Cause: RC defines the system error and the primitive. A system error.

Action: Call the Service Center.

NG061 SEVERITY 4 **NETGEN INTERNAL ERROR (GCL STRING)** *reason*

Cause: When present, *reason* is intended for detailed diagnosis. This is a NETGEN internal error, caused by the directive for which this error appears.

Action: Call the Service Center. A temporary solution is to suppress the corresponding directive.

NG062 SEVERITY 4 **ABNORMAL PROCESSING OF DICTIONARY:** CDH*i_dec2*

Cause: CDH*i_dec2* is intended for detailed diagnosis. Internal error from the communications dictionary access method.

Action: Call the Service Center.

NG063 SEVERITY 4 **NETGEN INTERNAL ERROR** *[reason]*

Cause: *reason* is intended only for detailed diagnosis. A system error in NETGEN.

Action: Call the Service Center.

NG064 SEVERITY 4 **INVALID OPTION STRING** *[reason]*

Cause: *reason* is intended for detailed diagnosis. A system error has occurred when starting NETGEN.

Action: Correct erroneous CRNETGEN parameters and rerun NETGEN. If the same error recurs, call the Field Service Center.

NG065 SEVERITY 4 **TELECOMMUNIC.SESSION IN PROGRESS->VCAM USERS**

NG065 SEVERITY 4 **TELECOMMUNIC. SESSION IN PROGRESS -> SERVERS**

Cause: The literal headed by -> specifies what kind of object has been found already active. An attempt has been made to run NETGEN with ENABLE while one of the following is still running:
- TNS, FEPS, RAEH, FECM and QMON
- any application using MAM.

This error condition is also reported at the operator terminal by the following message:

```
NG13 WARNING: SOME TELECOMMUNICATION OBJECTS
ARE STILL ACTIVE: GENERATION IS ABORTED.
```

Action: Wait for the objects indicated to complete, then rerun NETGEN.

NG066 SEVERITY 4 **UNABLE TO INITIALIZE PRIVATE DICTIONARY** RC=rc

Cause: RC defines the error and the primitive. NETGEN internal error.

Action: Call the Service Center.

NG067 SEVERITY 4 **UNABLE TO FORMAT QUEUE FILE** RC=rc

Cause: RC specifies the error and the system primitive. NETGEN has aborted while formatting the disk queue file.

Action: Rerun NETGEN. If the same failure recurs, reallocate the file:
- either to a different area of the same disk
- or on another disk.

NG068 SEVERITY 4 **ERROR DURING SRST FILE OPEN** RC=rc

Cause: RC specifies the error and the system primitive. The private SRST file specified cannot be opened.

Action: Check that the SRST file specified for SRSTFILE of the CRNETGEN command is present and available. Then rerun NETGEN.

NG069 SEVERITY 4 **ERROR WHEN ACCESSING SRST FILE** RC=rc

Cause: RC specifies the error and the system primitive. Records in the private SRST cannot be accessed.

Action: Rerun the NETGEN utility. If the same failure recurs, reallocate and rebuild the private SRST file on:
- either a different area of the same disk
- or another disk.

NG070 SEVERITY 4 **NETGEN ALREADY IN PROGRESS**

Cause: An attempt to run NETGEN (other than in simulation mode) has been made while another NETGEN (not in simulation mode) is already executing.

Action: Wait for the executing NETGEN to complete before resubmission.

NG071 SEVERITY 4 **UNABLE TO GET OPTION STRING** RC=rc

Cause: RC specifies the error and the system primitive. System error.

Action: Call the Service Center.

NG074 SEVERITY **XCP2WKS REFERENCED IN POOL MUST BE DECLARED WITH SYNCPOINT=1 :** *name1 / name2*

Cause: *name1* and *name2* identify the xcp2wks and pool objects declared respectively by a XCP2WKS and a POOL directive.
The XCP2WKS referenced in the POOL directive has the attribute SYNCPOINT=0 while the pool has the attribute SYNCLVL=SYNCPOINT.

Action: Modify either SYNCPOINT=1 in the directive XCP2WKS or SYNCLVL=CONFIRM in the directive POOL. Then rerun NETGEN.

NG075 SEVERITY **XCP2COR REFERENCED IN POOL MUST BE DECLARED WITH PARALLEL=1:** *name1 / name2*

Cause: *name1* and *name2* identify the xcp2 correspondent and pool objects declared respectively by a XCP2COR and a POOL directive.
The XCP2COR referenced in the POOL directive has the attribute PARALLEL=0 while the pool has the attribute MAXSESS > 1.

Action: Modify either PARALLEL=1 in the directive XCP2COR or MAXSESS <= 1 in the directive POOL. Then rerun NETGEN.

NG076 SEVERITY 3 **CONFLICT: CONFIG HOSTID NAME/LOC SYSTEM NAME**

Cause: The name for the LSYS parameter in CONFIG does not match the name for the LSYS directive in NETGEN.

Action: Correct, as appropriate and rerun NETGEN.

NG076 SEVERITY 4 **CONFLICT: CONFIG HOSTID NAME/LOC SYSTEM NAME**

Cause: The name for LSYS in CONFIG does not match LSYS in the Network Generation to be loaded.

Action: A Network Generation can only be loaded (SAVE = SYS or ENABLE) on the system for which it was prepared. Correct the name in the LSYS directive, then rerun NETGEN.

NG078 SEVERITY **NAME ALREADY USED** *name*

Cause: *name* identifies the duplicated name. The current object such as correspondent, logical application or workstation, has the same name as another object of the same type.

Action: Modify the directive(s) concerned and rerun NETGEN.

NG079 SEVERITY **ADDRESS ALREADY USED** *address*

Cause: *address* identifies the duplicated address. One of the following has been duplicated:
- either an XCP2 correspondent has a correspondent address (SESS + MAILBOX parameters) already used by another correspondent
- or a session control in the directory has a DSA address (SCID in the RSC directive) already used by another session control
- or a subdomain in the directory has an NSAP address already used by another suddomain.

Action: Modify the directive(s) concerned and rerun NETGEN.

NG080 SEVERITY 3 **MAILBOX ALREADY USED**

Cause: A mailbox name (XCP1MBX or MASTERMBX) is identical to the NAME parameter of this TDSWKS or XCP2WKS directive.

Action: Modify the directive(s) concerned and rerun NETGEN.

NG081 SEVERITY 4 **DIRECTORY OVERFLOW : TOO MANY CORRESPONDENTS DECLARED**

Cause: The maximum number of objects in the correspondent dictionary has been reached.

Action: Delete unused objects and rerun NETGEN.

NG081 SEVERITY 4 **DIRECTORY OVERFLOW : TOO MANY FORWARD REFERENCES FOR CORRESPONDENTS**

Cause: The maximum number of forward references for correspondents has been reached.
Note:
A forward reference occurs each time an object references another object which has not yet appeared in the configuration description.

Action: Rearrange the directives to reduce forward references, then resubmit NETGEN.

NG081 SEVERITY 4 **DIRECTORY OVERFLOW : TOO MANY REMOTE SYSTEMS DECLARED**

Cause: The maximum number of objects in the remote system dictionary has been reached.

Action: Delete unused objects and rerun NETGEN.

NG081 SEVERITY 4 **DIRECTORY OVERFLOW : TOO MANY FORWARD REFERENCES FOR REMOTE SYSTEMS**

Cause: The maximum number of forward references for remote systems has been reached.
Note:
A forward reference occurs each time an object references another object which has not yet appeared in the configuration description.

Action: Rearrange the directives to reduce forward references, then resubmit NETGEN.

NG082 SEVERITY 4 **UNABLE TO ENABLE NEW TERMINAL TABLE**

Cause: An abnormal condition has occurred while trying to enable the terminal table from the permanent workspace into the running workspace.

Action: Call the Service Center.

NG083 SEVERITY 4 **UNABLE TO EXPAND SEGMENT SIZE** *reason*

Cause: *reason* gives the diagnosis of the error condition which has occurred while trying to extend the segment size of either the communications root table or the VCAM main table.

Action: Call the Service Center.

NG084 SEVERITY 4 **UNABLE TO CREATE SEGMENT** *[reason]*

Cause: *reason* diagnoses the error condition which has occurred while creating data segments for internal use.

Action: Call the Service Center.

NG085 SEVERITY 4 **DISK QFILE NOT SPECIFIED**

Cause: Although enabling of tables is requested and disk file queues are used, no QFILE parameter has been declared in the CRNETGEN command.

Action: Allocate the file, add QFILE specifying the file in the CRNETGEN command to rerun NETGEN.

NG086 SEVERITY 4 **UNABLE TO ENABLE NETWORK TABLES** -> *reason* RC=rc

Cause: Abnormal system condition has occurred while trying to enable the network tables from the permanent workspace into the running workspace.

Action: Call the Service Center.

NG087 SEVERITY 4 **UNABLE TO CREATE TERMINAL TABLE**

Cause: Abnormal error condition has occurred while trying to create a new terminal table.

Action: Check that the following members are present and correct:
- H_TERM in the library SYS.HSLLIB
- and that specified for CRTERM in the CRNETGEN command.

If yes, call the Service Center.

NG088 SEVERITY 4 **UNABLE TO ABSOLUTIZE ADDRESSES** *[reason]*

Cause: *reason* is a diagnostic aid for detailed analysis. A system internal error has occurred while trying to build link addresses between tables.

Action: Call the Service Center.

NG089 SEVERITY 4 **UNABLE TO INITIALIZE ADMINISTRATION TABLES**

Cause: A system error has occurred while trying to initialize working tables for network administration.

Action: Call the Service Center.

NG090 SEVERITY 2 **MAXIMUM SESSIONS REDUCED TO 1.**

Cause: The maximum number of sessions defined by MAXSESS in the AG directive has been reduced to 1 because of backup ACs and AG have been declared INIT=AF.

Action: None

NG091 SEVERITY 2 **IADDR OUT OF RANGE OF LPL RADDR**

Cause: Ethernet address (IADDR) declared in the directive RPL is out of range(s) of RADDR declared in the directive LPL.

Action: Modify the range(s) of RADDR declared in the directive LPL.

NG092 SEVERITY 2 **PID (ISL1) NOT AVAILABLE ON YOUR SITE ANY CONECTION COULD BE REJECTED**

Cause: The MI for ISL1 is not available.

Action: Request the MI for ISL1.

NG094 SEVERITY 2 **A SVR FEPS IS NOT REF IN A RSYS:** *name*

Cause: An FEPS type server explicitly declared in an SVR directive is not referenced by any RSYS directive.

Action: This may be perfectly normal for some networks. If this is the case, ignore this message.
If the reference has been omitted, correct the appropriate RSYS directive and rerun NETGEN.

NG098 SEVERITY **LCT NAME OR LT NUMBER NOT FOUND OR WRONG IN SRST:** *name*

Cause: An LCT or LT number specified in the network description is not found in the (local or remote) SRST table.

Action: If the SRST file is that expected for this configuration, correct these values and rerun NETGEN. Otherwise, resubmit NETGEN specifying the appropriate SRST file.

NG099 SEVERITY **DUPLICATED DTE ADDRESS:** *address*

Cause: *address* for the DTE is not unique in the network description.

Action: Correct the duplicated DTE addresses and rerun NETGEN.

NG100 SEVERITY 4 **=====> EXCEPTION *id* AT ADDRESS** *address*

Cause: *id* and *address* indicate the exception condition and its address. A system exception has occurred.

Action: Call the Service Center.

NG101 SEVERITY 3 **NO VALID TELECOMMUNICATION SESSION RUNNING**

Cause: No valid communications session is currently running, so incremental generation cannot be performed.

Action: Rerun NETGEN with CRNET specifying a basic network generation.

NG102 SEVERITY 3 **ERROR DURING DIRECTORY WORKSPACE HANDLING :** [reason]

Cause: *reason* ia a diagnostic aid for detailed analysis. A system error has occurred with the permanent workspace when enabling the directory configuration.

Action: Call the Service Center.

NG103 SEVERITY 3 **ERROR DURING DIRECTORY HANDLING :** *[reason]*

Cause: *reason* gives the diagnostic for field engineering personnel. A system error has occurred when handling an internal directory copy which serves as the intermediate exchange area between the permanent and the running workspaces.

Action: Call the Service Center.

NG104 SEVERITY **WRONG MUTUAL REFERENCING FOR BACKUP CORRESPONDENTS :** *name1* AND *name2*

Cause: *name1* and *name2* are the names specified for the NAME and BACKUP parameters of either a TMCOR, XCP1COR or XCP2COR directive.
The correspondent and its backup must be mutually declared for cross-reference.

Action: Correct the directives concerned and rerun NETGEN.

NG105 SEVERITY 2 **BACKUP ATTRIBUTES MISMATCH : FIRST=1 ASSUMED FOR** *name1* **AND FIRST=0 ASSUMED FOR** *name2*

Cause: *name1* and *name2* identify two mutual backup correspondents assumed by default to be primary (FIRST=1).
Mutual backup correspondents cannot both have the same FIRST attribute.

Action: If the default values assumed are not satisfactory, modify the directive(s) concerned and rerun NETGEN.

NG105 SEVERITY 2 **BACKUP ATTRIBUTES MISMATCH: PROFILE OF** *name1* **ASSUMED FOR** *name2*

Cause: *name1* and *name2* are correspondents declared as mutual backups and having respectively FIRST=1 and FIRST=0.
The two correspondents have different profiles (PARALLEL + PRIMARY + WINNER + SYNCLVL + USERCTL + PROJCTL).

Action: If the defaults assumed are not satisfactory, correct the directive(s) concerned and rerun NETGEN.

NG106 SEVERITY **SYSTEM NOT DECLARED FOR THIS SC:** *name*

Cause: *name* gives the session control name in error. An RSC directive has been specified in the directory configuration but is not referenced by any RSYS directive.

Action: Correct the relevant directive(s) and rerun NETGEN.

NG106 SEVERITY **SYSTEM NOT DECLARED:** *name*

Cause: *name* specifies the system in error. The system specified in a SDOM directive has not been declared in the basic network configuration description.

Action: Insert the misssing directive and rerun NETGEN.

NG107 SEVERITY 3 **TRANSPORT STATION NOT DECLARED AS RTS OF PASSTHROUGH** SYSTEM: *name*

Cause: The transport station *name* in the list of transport stations of a PASS-THROUGH RSC directive has not been declared by an RTS directive: or this RTS is not declared as a transport station of a PASS-THROUGH RSC directive (RSC TYPE=PT) in the basic network configuration description.

Action: Either declare the transport station in an RTS directive or refer to an RTS already declared as a transport station in a PASS-THROUGH RSC directive, then rerun NETGEN.

NG108 SEVERITY **SERVER NOT DECLARED :** *name*

Cause: *name* identifies the server. The server specified either in the list of servers of an RSC directive or in an SR directive, has not been declared by an SVR directive in the basic network configuration description.

Action: Either insert an SVR directive or remove the server from the list in the RSC directive. Then rerun NETGEN.

NG109 SEVERITY **SYSTEM LISTED IN SDOM IS NEITHER PASS-THROUGH NOR NEIGHBOUR :** *name*

Cause: *name* identifies the remote system. A system listed in an SDOM directive is declared neither passthrough nor neighbor in the basic network configuration description.

Action: Correct the directive concerned and rerun NETGEN.

NG112 SEVERITY 3 **XCP2 WORKSTATION NOT REFERENCED BY ANY XCP2 CORRESPONDENT :** *name*

Cause: *name* identifies the workstation. The specified XCP2 workstation is not referenced by any XCP2COR directive using either the workstation or its mailbox with the name in the LSC directive.

Action: Correct or add the relevant directives and rerun NETGEN.

NG113 SEVERITY 4 **CURRENT GCL DOMAIN VERSION NOT COMPATIBLE WITH THAT SAVED**

Cause: The GCL environment version used when compiling the currently running basic network configuration is no longer compatible with that currently used for compiling the directory configuration description.

Action: Recompile the basic network configuration description before or with the directory configuration description.

NG114 SEVERITY 4 **ABNORMAL RESTORE OF GCL ENVIRONMENT FOR :** *[name] -> reason*

Cause: *name* identifies the element of the GCL environment. *reason* is intended for detailed diagnosis.
A system error has occurred while restoring for an incremental generation, the whole GCL environment as it was at the end of the corresponding basic network generation.

Action: Call the Service Center.

NG115 SEVERITY **TOO MANY CORRESPONDENTS DECLARED WITHIN TCORR DICTIONARY**

Cause: The maximum number of correspondents in the directory configuration has been exceeded. See limits in Section 10.

Action: Call the Service Center.

NG116 SEVERITY **TOO MANY REMOTE SYSTEMS DECLARED WITHIN TSYS DICTIONARY**

Cause: The maximum number of remote systems in the directory configuration has been exceeded. See limits in Section 10.

Action: Move those most frequently accessed to the basic network configuration, then rerun NETGEN. Otherwise call the Service Center.

NG117 SEVERITY **TOO MANY REMOTE SESSION CONTROL DECLARED WITHIN TSYS DICTIONARY**

Cause: The maximum number of session control in the directory configuration has been exceeded. See limits in Section 10.

Action: Move some of them to the basic network configuration and rerun NETGEN. Otherwise call the Service Center.

NG118 SEVERITY 4 **NO VALID REMOTE SYSTEM DIRECTORY RUNNING: INCREMENTAL GENERATION ABORTED**

Cause: The currently running directory is not valid probably because a previous incremental generation has already aborted. No incremental generation is possible.

Action: Rerun NETGEN for a full generation :basic + directory.

NG120 SEVERITY 3 **TYPE CONFLICT BETWEEN SESSION CONTROL AND SYSTEM :** *name*

Cause: *name* specified for TYPE in the RSC directive is different from that specified with the corresponding RSYS directive.

Action: Correct the directive concerned and rerun NETGEN.

NG121 SEVERITY 3 **ADDRESSING TYPE CONFLICT BETWEEN SESSION CONTROL AND SYSTEM :** *name*

Cause: *name* specified for ADDRTYPE in the RSC directive is different from that specified in the corresponding RSYS directive.

Action: Correct the directive concerned and rerun NETGEN.

NG122 SEVERITY 3 **RTS NAME MISMATCH BETWEEN SESSION ROUTE AND SESSION CONTROL :** *name1 name2*

Cause: *name1* the RTS is associated with *name2* its session control in the SR directive, but a corresponding RSC directive does not have the RTS in its RTS list.

Action: Correct the directive(s) concerned and rerun NETGEN.

NG123 SEVERITY 3 **SVR NAME MISMATCH BETWEEN SESSION ROUTE AND SESSION CONTROL :** *name1 name2*

Cause: *name1* the SVR is associated with *name2* the session control in an SR directive, but the corresponding RSC directive does not have the SVR in its SVR list.

Action: Correct the directive(s) concerned and rerun NETGEN.

NG124 SEVERITY **NOT DECLARED SESSION CONTROL REFERENCED BY A SESSION ROUTE :** *name*

Cause: *name* identifies the session control referenced in an SR directive but is not declared in an RSC directive.

Action: Insert an RSC directive and rerun NETGEN.

NG130 SEVERITY {3|2} **TOO MANY STATISTICS BLOCKS DECLARED.**

Cause: Overflow on the table which contains the statistics blocks:
- if severity=3, overflow has occurred with SBs explicitly declared
- if severity=2, some default SBs cannot be entered in the table.

Action: If severity=3, reduce the number of SB directives so that the total number of SBs (explicit and default) are under the limit given in Section 10. Then rerun NETGEN.

NG131 SEVERITY 3 **OBJECT NOT DECLARED (USED IN SB):** *type*: *name*

Cause: *type* and *name* identify the object referenced by an SB directive but is not present in the basic network configuration.

Action: Correct or remove the SB directive concerned from the basic network configuration description and rerun NETGEN.

NG132 SEVERITY 2 **REDUNDANT SB DECLARATION:** *name1 name2*

Cause: *name1* and *name2* identify the same connection object.

Action: Depending on the case:
- either rerun NETGEN after suppressing one of the redundant SB in the basic network configuration description
- or avoid during subsequent communications sessions setting the both SBs to *enabled*.

NG133 SEVERITY 4 **UNABLE TO ENABLE STATISTICS BLOCKS [-> FOR SB TABLE]** rc

Cause: An error flagged by the rc, has occurred while enabling the statistics blocks table.

Action: Depending on the rc, either rerun NETGEN or call the Service Center.

B. Server Job Occurrence Reports

B.1 OCS (Open Communications Subsystem)

In OCS, the process numbers are:

Drivers

Decimal	**Hexadecimal**	**Process**
0	0	NCCD7 Driver
5	5	IPSS (IPS Server)
6	6	ADMA (DSAC Administration)
15	F	LIPS (Local IPS Administration)
16	10	SGMR (SNMP Manager)
25	19	STRC (IPS Trace)

Servers

Decimal	**Hexadecimal**	**Process**
1,2,3,4	1,2,3,4	PPAS (OSI Server)
7,8,9,10	7,8,9,A	CONF (Configuration)
11,12,13,14	B,C,D,E	ITRC (RST ISO Trace)
17,18,19,20	11,12,13,14	PTRC (PPA Trace)
21,22,23,24	15,16,17,18	SNMP (SNMP Agent)

B.1.1 JOR (Job Occurrence Report)

In the OCS JOR given on the following pages, spaces have been suppressed or added to improve legibility. Information which is not essential to the discussion is not shown and is replaced by dots (...).

```
JOBID=OCS  USER=OP5  PROJECT=OPERATOR  BILLING=INSTALL  RON=X1564
--------------------------------------------------------------------------------
09:18:12  JOB EXECUTION LISTING                                     MAY 07, 1996
--------------------------------------------------------------------------------
STEP 1
LOAD MODULE = H_NCCD7 (17:42 APR 29, 1996 )
LIBRARY = SYS.HLMLIB

09:18:26
STEP STARTED XPRTY=0 (MAY 07, 1996)
CLR00: C Run Time version 40.00
CLR00: C Run Time version 40.00
CLR00: C Run Time version 40.00

**************************************************************
**              X B F   S T A T I S T I C S                 **
**************************************************************
 ---> BUFFER POOL CREATION FOR NCC1
09:18:35
BUFFER UNIT SIZE                                     :      256
MAXIMUM NUMBER OF BUFFER UNIT                        :    32768
MINIMUM NUMBER OF BUFFER UNIT                        :     2048
CURRENT NUMBER OF SEGMENT BUFFER                     :        8

LTH: 09:18:35 MAY 07, 1996 0001-PPAS v_op DEACT_LT LT31 1.ALT 0 0 LTH_LTH SYNC_OUT
LTH: 09:18:35 MAY 07, 1996 0001-PPAS lt_init DEACT_LT LT31 1.ALT 0 0 NCC1 EA31 LT31

CLR00: C Run Time version 40.00
CLR00: C Run Time version 40.00
CLR00: C Run Time version 40.00
CLR00: C Run Time version 40.00
CLR00: C Run Time version 40.00
CLR00: C Run Time version 40.00

LTH: 09:18:36 MAY 07, 1996 0001-PPAS warni-40 MESSAGE LT31 1.ALT 13 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:18:36 MAY 07, 1996 0001-PPAS v_op MESSAGE LT31 1.ALT 13 0 LTH_STARTER LT_KO-host
LTH: 09:18:36 MAY 07, 1996 0001-PPAS warni-8A MESSAGE LT31 1.ALT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:18:36 MAY 07, 1996 0001-PPAS lt_init ACTIV_LT LT31 1.ALT 0 0 NCC1 EA31 LT31
```

```
LTH: 09:18:38 MAY 07, 1996 0001-PPAS v_op MESSAGE LT31 1.ALT 3 4 LTH_STARTER LT_OK

 NC06 FROM SERVER NCC1 LOAD STARTED FROM VERSION: N206

 NC05 FROM SERVER NCC1 LOAD COMPLETED

LTH: 09:18:47 MAY 07, 1996 0001-PPAS warni-3F MESSAGE LT31 1.ALT 4 4
(TERM_ATT FIFO_RDY ) attention received
RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:18:48 MAY 07, 1996 0001-PPAS v_op MESSAGE LT31 1.ALT 4 4 LTH_STARTER PPA_READY
LTH: 09:18:48 MAY 07, 1996 0001-PPAS warni-56 MESSAGE LT31 1.ALT 4 4
(TERM_ATT PPA_RDY ) attention received
RC=148A0015->NCC 10,ALMOST
LTH: 09:18:49 MAY 07, 1996 0001-PPAS v_op DEACT_LT LT32 1.ISOLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:18:49 MAY 07, 1996 0001-PPAS lt_init DEACT_LT LT32 1.ISOLT 0 0 NCC1 EA31 LT32
LTH: 09:18:49 MAY 07, 1996 0011-PTRC v_op DEACT_LT LT35 1.TRCLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:18:49 MAY 07, 1996 0011-PTRC lt_init DEACT_LT LT35 1.TRCLT 0 0 NCC1 EA31 LT35
LTH: 09:18:49 MAY 07, 1996 0011-PTRC warni-40 MESSAGE LT35 1.TRCLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:18:49 MAY 07, 1996 0011-PTRC v_op MESSAGE LT35 1.TRCLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:18:49 MAY 07, 1996 0011-PTRC warni-8A MESSAGE LT35 1.TRCLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:18:49 MAY 07, 1996 0011-PTRC lt_init ACTIV_LT LT35 1.TRCLT 0 0 NCC1 EA31 LT35
LTH: 09:18:49 MAY 07, 1996 0011-PTRC v_op MESSAGE LT35 1.TRCLT 1 4 LTH_STARTER LT_OK
........................
........................
LTH: 09:18:50 MAY 07, 1996 0001-PPAS warni-40 MESSAGE LT32 1.ISOLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:18:50 MAY 07, 1996 0001-PPAS v_op MESSAGE LT32 1.ISOLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:18:50 MAY 07, 1996 0001-PPAS warni-8A MESSAGE LT32 1.ISOLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:18:50 MAY 07, 1996 0001-PPAS lt_init ACTIV_LT LT32 1.ISOLT 0 0 NCC1 EA31 LT32
LTH: 09:18:54 MAY 07, 1996 0001-PPAS v_op MESSAGE LT32 1.ISOLT 2 4 LTH_STARTER LT_OK

 NC01 FROM SERVER NCC1 : ISO/DIWS+IPS  PROTOCOLS AVAILABLE

LTH: 09:18:57 MAY 07, 1996 0005-IPSS v_op DEACT_LT LT33 1.IPSLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:18:57 MAY 07, 1996 0005-IPSS lt_init DEACT_LT LT33 1.IPSLT 0 0 NCC1 EA31 LT33
LTH: 09:18:57 MAY 07, 1996 0005-IPSS warni-40 MESSAGE LT33 1.IPSLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:18:57 MAY 07, 1996 0005-IPSS v_op MESSAGE LT33 1.IPSLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:18:57 MAY 07, 1996 0005-IPSS warni-8A MESSAGE LT33 1.IPSLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:18:57 MAY 07, 1996 0005-IPSS v_op DEACT_LT LT53 2.IPSLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:18:57 MAY 07, 1996 0005-IPSS lt_init DEACT_LT LT53 2.IPSLT 0 0 FREE_NCC EA51 LT53
```

```
LTH: 09:18:57 MAY 07, 1996 0005-IPSS warni-40 MESSAGE LT53 2.IPSLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:18:57 MAY 07, 1996 0005-IPSS v_op MESSAGE LT53 2.IPSLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:18:57 MAY 07, 1996 0005-IPSS warni-8A MESSAGE LT53 2.IPSLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:18:58 MAY 07, 1996 0005-IPSS v_op DEACT_LT LT33 1.IPSLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:18:58 MAY 07, 1996 0005-IPSS lt_init DEACT_LT LT33 1.IPSLT 0 0 NCC1 EA31 LT33
LTH: 09:18:58 MAY 07, 1996 0005-IPSS warni-40 MESSAGE LT33 1.IPSLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:18:58 MAY 07, 1996 0005-IPSS v_op MESSAGE LT33 1.IPSLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:18:58 MAY 07, 1996 0005-IPSS warni-8A MESSAGE LT33 1.IPSLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:18:58 MAY 07, 1996 0005-IPSS lt_init ACTIV_LT LT33 1.IPSLT 0 0 NCC1 EA31 LT33
LTH: 09:18:58 MAY 07, 1996 0005-IPSS v_op MESSAGE LT33 1.IPSLT 2 4 LTH_STARTER LT_OK
LTH: 09:18:59 MAY 07, 1996 0001-PPAS v_op MESSAGE LT31 1.ALT 9 4 LTH_STARTER TRSV_READY

 NC01 FROM SERVER NCC1 : CONNECTED TO NETWORK

LTH: 09:19:02 MAY 07, 1996 0015-SNMP v_op DEACT_LT LT34 1.SNMPLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:19:02 MAY 07, 1996 0015-SNMP lt_init DEACT_LT LT34 1.SNMPLT 0 0 NCC1 EA31 LT34
LTH: 09:19:02 MAY 07, 1996 0015-SNMP warni-40 MESSAGE LT34 1.SNMPLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:19:02 MAY 07, 1996 0015-SNMP v_op MESSAGE LT34 1.SNMPLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:19:02 MAY 07, 1996 0015-SNMP warni-8A MESSAGE LT34 1.SNMPLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:19:02 MAY 07, 1996 0015-SNMP lt_init ACTIV_LT LT34 1.SNMPLT 0 0 NCC1 EA31 LT34
LTH: 09:19:02 MAY 07, 1996 0015-SNMP v_op MESSAGE LT34 1.SNMPLT 1 4 LTH_STARTER LT_OK
........................
........................

**************************************************************
**               X B F    S T A T I S T I C S             **
**************************************************************
 ---> BUFFER POOL CREATION FOR NCC2
09:19:33
BUFFER UNIT SIZE                                     :      256
MAXIMUM NUMBER OF BUFFER UNIT                        :    32768
MINIMUM NUMBER OF BUFFER UNIT                        :     2048
CURRENT NUMBER OF SEGMENT BUFFER                     :        8

LTH: 09:19:33 MAY 07, 1996 0002-PPAS v_op DEACT_LT LT51 2.ALT 0 0 LTH_LTH SYNC_OUT
LTH: 09:19:33 MAY 07, 1996 0002-PPAS lt_init DEACT_LT LT51 2.ALT 0 0 NCC2 EA51 LT51
LTH: 09:19:34 MAY 07, 1996 0002-PPAS warni-40 MESSAGE LT51 2.ALT 13 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
```

```
LTH: 09:19:34 MAY 07, 1996 0002-PPAS v_op MESSAGE LT51 2.ALT 13 0 LTH_STARTER LT_KO-host
LTH: 09:19:34 MAY 07, 1996 0002-PPAS warni-8A MESSAGE LT51 2.ALT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:19:34 MAY 07, 1996 0002-PPAS lt_init ACTIV_LT LT51 2.ALT 0 0 NCC2 EA51 LT51
LTH: 09:19:36 MAY 07, 1996 0002-PPAS v_op MESSAGE LT51 2.ALT 3 4 LTH_STARTER LT_OK

 NC06 FROM SERVER NCC2 LOAD STARTED FROM VERSION: N206

 NC05 FROM SERVER NCC2 LOAD COMPLETED

LTH: 09:19:45 MAY 07, 1996 0002-PPAS warni-3F MESSAGE LT51 2.ALT 4 4
(TERM_ATT FIFO_RDY ) attention received
RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:19:46 MAY 07, 1996 0002-PPAS v_op MESSAGE LT51 2.ALT 4 4 LTH_STARTER PPA_READY
LTH: 09:19:46 MAY 07, 1996 0002-PPAS warni-56 MESSAGE LT51 2.ALT 4 4
(TERM_ATT PPA_RDY ) attention received
RC=148A0015->NCC 10,ALMOST
LTH: 09:19:47 MAY 07, 1996 0002-PPAS v_op DEACT_LT LT52 2.ISOLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:19:47 MAY 07, 1996 0002-PPAS lt_init DEACT_LT LT52 2.ISOLT 0 0 NCC2 EA51 LT52
LTH: 09:19:47 MAY 07, 1996 0012-PTRC v_op DEACT_LT LT55 2.TRCLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:19:47 MAY 07, 1996 0012-PTRC lt_init DEACT_LT LT55 2.TRCLT 0 0 NCC2 EA51 LT55
LTH: 09:19:47 MAY 07, 1996 0012-PTRC warni-40 MESSAGE LT55 2.TRCLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:19:47 MAY 07, 1996 0012-PTRC v_op MESSAGE LT55 2.TRCLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:19:47 MAY 07, 1996 0012-PTRC warni-8A MESSAGE LT55 2.TRCLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:19:47 MAY 07, 1996 0012-PTRC lt_init ACTIV_LT LT55 2.TRCLT 0 0 NCC2 EA51 LT55
LTH: 09:19:47 MAY 07, 1996 0012-PTRC v_op MESSAGE LT55 2.TRCLT 1 4 LTH_STARTER LT_OK
........................
........................
LTH: 09:19:48 MAY 07, 1996 0002-PPAS warni-40 MESSAGE LT52 2.ISOLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:19:48 MAY 07, 1996 0002-PPAS v_op MESSAGE LT52 2.ISOLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:19:48 MAY 07, 1996 0002-PPAS warni-8A MESSAGE LT52 2.ISOLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:19:48 MAY 07, 1996 0002-PPAS lt_init ACTIV_LT LT52 2.ISOLT 0 0 NCC2 EA51 LT52
LTH: 09:19:52 MAY 07, 1996 0002-PPAS v_op MESSAGE LT52 2.ISOLT 2 4 LTH_STARTER LT_OK

 NC01 FROM SERVER NCC2 : ISO/DIWS+IPS  PROTOCOLS AVAILABLE

LTH: 09:19:55 MAY 07, 1996 0005-IPSS v_op DEACT_LT LT53 2.IPSLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:19:55 MAY 07, 1996 0005-IPSS lt_init DEACT_LT LT53 2.IPSLT 0 0 NCC2 EA51 LT53
LTH: 09:19:55 MAY 07, 1996 0005-IPSS warni-40 MESSAGE LT53 2.IPSLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:19:55 MAY 07, 1996 0005-IPSS v_op MESSAGE LT53 2.IPSLT 8 0 LTH_STARTER LT_KO-host
```

```
LTH: 09:19:55 MAY 07, 1996 0005-IPSS warni-8A MESSAGE LT53 2.IPSLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:19:55 MAY 07, 1996 0005-IPSS lt_init ACTIV_LT LT53 2.IPSLT 0 0 NCC2 EA51 LT53
LTH: 09:19:55 MAY 07, 1996 0005-IPSS v_op MESSAGE LT53 2.IPSLT 2 4 LTH_STARTER LT_OK
LTH: 09:19:56 MAY 07, 1996 0002-PPAS v_op MESSAGE LT51 2.ALT 9 4 LTH_STARTER TRSV_READY

 NC01 FROM SERVER NCC2 : CONNECTED TO NETWORK

LTH: 09:19:58 MAY 07, 1996 0016-SNMP v_op DEACT_LT LT54 2.SNMPLT 0 0 LTH_LTH SYNC_OUT
LTH: 09:19:58 MAY 07, 1996 0016-SNMP lt_init DEACT_LT LT54 2.SNMPLT 0 0 NCC2 EA51 LT54
LTH: 09:19:58 MAY 07, 1996 0016-SNMP warni-40 MESSAGE LT54 2.SNMPLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:19:58 MAY 07, 1996 0016-SNMP v_op MESSAGE LT54 2.SNMPLT 8 0 LTH_STARTER LT_KO-host
LTH: 09:19:58 MAY 07, 1996 0016-SNMP warni-8A MESSAGE LT54 2.SNMPLT 0 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:19:58 MAY 07, 1996 0016-SNMP lt_init ACTIV_LT LT54 2.SNMPLT 0 0 NCC2 EA51 LT54
LTH: 09:19:58 MAY 07, 1996 0016-SNMP v_op MESSAGE LT54 2.SNMPLT 1 4 LTH_STARTER LT_OK

 NC01 FROM SERVER NCC1 : SERVER STOPPING

LTH: 09:22:05 MAY 07, 1996 0005-IPSS v_op DEACT_LT LT33 1.IPSLT 4 4 LTH_LTH SYNC_OUT
LTH: 09:22:05 MAY 07, 1996 0005-IPSS warni-41 MESSAGE LT33 1.IPSLT 8 0
(DATA_OUT abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:05 MAY 07, 1996 0005-IPSS warni-40 MESSAGE LT33 1.IPSLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:22:05 MAY 07, 1996 0005-IPSS warni-8A MESSAGE LT33 1.IPSLT 9 0
(DATA_IN abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:05 MAY 07, 1996 0005-IPSS warni-8A MESSAGE LT33 1.IPSLT 9 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:05 MAY 07, 1996 0005-IPSS v_op MESSAGE LT33 1.IPSLT 9 0 LTH_STARTER LT_KO-host

 NC01 FROM SERVER NCC1 : IPS TRAFFIC STOPPED

 NC01 FROM SERVER NCC1 : NO MORE TRAFFIC

LTH: 09:22:06 MAY 07, 1996 0011-PTRC v_op DEACT_LT LT35 1.TRCLT 4 5 LTH_LTH SYNC_OUT
LTH: 09:22:06 MAY 07, 1996 0001-PPAS v_op DEACT_LT LT32 1.ISOLT 4 4 LTH_LTH SYNC_OUT
>>> IFN =EA310003
    REWRITECNT =        0    DELETECNT  =        0    WRITECNT   =        0    READCNT    =  0
    EFN =SYS.DSATRC_EA31_3
    GETCICOUNT =       20    HITCOUNT   =       17    IOCOUNT    =       10
```

```
LTH: 09:22:06 MAY 07, 1996 0011-PTRC warni-40 MESSAGE LT35 1.TRCLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
........................
........................
LTH: 09:22:07 MAY 07, 1996 0011-PTRC warni-8A MESSAGE LT35 1.TRCLT 9 0
(TERM_IN abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:07 MAY 07, 1996 0011-PTRC warni-8A MESSAGE LT35 1.TRCLT 9 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:07 MAY 07, 1996 0011-PTRC v_op MESSAGE LT35 1.TRCLT 9 0 LTH_STARTER LT_KO-host
LTH: 09:22:07 MAY 07, 1996 0001-PPAS warni-41 MESSAGE LT32 1.ISOLT 8 0
(DATA_OUT abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:07 MAY 07, 1996 0001-PPAS warni-40 MESSAGE LT32 1.ISOLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:22:07 MAY 07, 1996 0001-PPAS warni-8A MESSAGE LT32 1.ISOLT 9 0
(DATA_IN abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:07 MAY 07, 1996 0001-PPAS warni-8A MESSAGE LT32 1.ISOLT 9 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:08 MAY 07, 1996 0001-PPAS v_op MESSAGE LT32 1.ISOLT 9 0 LTH_STARTER LT_KO-host
LTH: 09:22:08 MAY 07, 1996 0001-PPAS v_op DEACT_LT LT31 1.ALT 12 4 LTH_LTH SYNC_OUT
LTH: 09:22:09 MAY 07, 1996 0001-PPAS warni-40 MESSAGE LT31 1.ALT 13 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:22:09 MAY 07, 1996 0001-PPAS warni-8A MESSAGE LT31 1.ALT 14 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:10 MAY 07, 1996 0001-PPAS v_op MESSAGE LT31 1.ALT 14 0 LTH_STARTER LT_KO-host

 ---> BUFFER POOL TERMINATION FOR NCC1
09:22:10
NUMBER OF BUFFER UNIT                                :     2048
NUMBER OF BUFFER UNIT USED                           :        0
EXTENSION THRESHOLD                                  :     1638
CURRENT NUMBER OF SEGMENT BUFFER                     :        8
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER         :      171
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER         :        0
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER         :        0
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER         :       57

  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER         :      140
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER         :        0
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER         :        0
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER         :        0
```

```
*************************************************************
**      E N D   O F   X B F   S T A T I S T I C S          **
*************************************************************

 NC01 FROM SERVER NCC2 : SERVER STOPPING

LTH: 09:22:23 MAY 07, 1996 0005-IPSS v_op DEACT_LT LT53 2.IPSLT 4 4 LTH_LTH SYNC_OUT
LTH: 09:22:23 MAY 07, 1996 0005-IPSS warni-41 MESSAGE LT53 2.IPSLT 8 0
(DATA_OUT abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:23 MAY 07, 1996 0005-IPSS warni-40 MESSAGE LT53 2.IPSLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:22:23 MAY 07, 1996 0005-IPSS warni-8A MESSAGE LT53 2.IPSLT 9 0
(DATA_IN abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:23 MAY 07, 1996 0005-IPSS warni-8A MESSAGE LT53 2.IPSLT 9 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:23 MAY 07, 1996 0005-IPSS v_op MESSAGE LT53 2.IPSLT 9 0 LTH_STARTER LT_KO-host

 NC01 FROM SERVER NCC2 : IPS TRAFFIC STOPPED

 NC01 FROM SERVER NCC2 : NO MORE TRAFFIC

LTH: 09:22:24 MAY 07, 1996 0012-PTRC v_op DEACT_LT LT55 2.TRCLT 4 5 LTH_LTH SYNC_OUT
LTH: 09:22:25 MAY 07, 1996 0002-PPAS v_op DEACT_LT LT52 2.ISOLT 4 4 LTH_LTH SYNC_OUT
........................
........................
LTH: 09:22:25 MAY 07, 1996 0012-PTRC warni-40 MESSAGE LT55 2.TRCLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
........................
........................
LTH: 09:22:25 MAY 07, 1996 0012-PTRC warni-8A MESSAGE LT55 2.TRCLT 9 0
(TERM_IN abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:25 MAY 07, 1996 0012-PTRC warni-8A MESSAGE LT55 2.TRCLT 9 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:25 MAY 07, 1996 0012-PTRC v_op MESSAGE LT55 2.TRCLT 9 0 LTH_STARTER LT_KO-host
LTH: 09:22:26 MAY 07, 1996 0002-PPAS warni-41 MESSAGE LT52 2.ISOLT 8 0
(DATA_OUT abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:26 MAY 07, 1996 0002-PPAS warni-40 MESSAGE LT52 2.ISOLT 8 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)
LTH: 09:22:26 MAY 07, 1996 0002-PPAS warni-8A MESSAGE LT52 2.ISOLT 9 0
(DATA_IN abnormal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
```

```
LTH: 09:22:26 MAY 07, 1996 0002-PPAS warni-8A MESSAGE LT52 2.ISOLT 9 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:26 MAY 07, 1996 0002-PPAS v_op MESSAGE LT52 2.ISOLT 9 0 LTH_STARTER LT_KO-host
LTH: 09:22:26 MAY 07, 1996 0002-PPAS v_op DEACT_LT LT51 2.ALT 12 4 LTH_LTH SYNC_OUT
LTH: 09:22:27 MAY 07, 1996 0002-PPAS warni-40 MESSAGE LT51 2.ALT 13 0
(LTH_LTH SYNC_OUT ) synchronisation
message received RC=148A0015->NCC 10,ALMOST (exit)

LTH: 09:22:27 MAY 07, 1996 0002-PPAS warni-8A MESSAGE LT51 2.ALT 14 0
(TERM_IN normal ) message ignored
RC=148A0005->NCC 10,IGNORE (exit)
LTH: 09:22:28 MAY 07, 1996 0002-PPAS v_op MESSAGE LT51 2.ALT 14 0 LTH_STARTER LT_KO-host

 ---> BUFFER POOL TERMINATION FOR NCC2
09:22:28
NUMBER OF BUFFER UNIT                                  :     2048
NUMBER OF BUFFER UNIT USED                             :        0
EXTENSION THRESHOLD                                    :     1638
CURRENT NUMBER OF SEGMENT BUFFER                       :        8
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :      171
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :        0
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :        0
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :      141
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :      122
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :        0
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :        0
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :        0
**************************************************************
**      E N D   O F   X B F   S T A T I S T I C S          **
**************************************************************

TASK MAIN PGID=0088 PRID=00 COMPLETED
TASK PPAS PGID=0088 PRID=01 ABORTED BY SYSTEM....
........................
TASK PPAS PGID=0088 PRID=02 ABORTED BY SYSTEM....
........................
TASK IPSS PGID=0088 PRID=05 ABORTED BY SYSTEM....
........................
TASK ADMA PGID=0088 PRID=06 ABORTED BY SYSTEM....
TASK CONF PGID=0088 PRID=07 ABORTED BY SYSTEM....
........................
TASK CONF PGID=0088 PRID=08 ABORTED BY SYSTEM....
........................
TASK ITRC PGID=0088 PRID=0B ABORTED BY SYSTEM....
TASK ITRC PGID=0088 PRID=0C ABORTED BY SYSTEM....
TASK LIPS PGID=0088 PRID=0F ABORTED BY SYSTEM....
TASK SMGR PGID=0088 PRID=10 ABORTED BY SYSTEM....
TASK PTRC PGID=0088 PRID=11 ABORTED BY SYSTEM....
TASK PTRC PGID=0088 PRID=12 ABORTED BY SYSTEMR...
TASK SNMP PGID=0088 PRID=15 ABORTED BY SYSTEM....
```

```
TASK SNMP PGID=0088 PRID=16 ABORTED BY SYSTEM....
TASK STRC PGID=0088 PRID=19 ABORTED BY SYSTEM....
........................
FP07.IFN:STDOUT   HAS BEEN CLOSED BY SYSTEM
FP07.IFN:STDERR   HAS BEEN CLOSED BY SYSTEM
SYSBKST   ON  S72B   :  NB  OF  IO  REQUESTS=            0
SYSPVMF   ON  ****** :  NB  OF  IO  REQUESTS=            9
SYSLIB    ON  ****** :  NB  OF  IO  REQUESTS=            0
SYSBKST*  ON  ****** :  NB  OF  IO  REQUESTS=            0
H_CONF1   ON  H72B   :  NB  OF  IO  REQUESTS=            6
H_SLIB1   ON  H72B   :  NB  OF  IO  REQUESTS=            2
H_BLIB1   ON  H72B   :  NB  OF  IO  REQUESTS=          768
3-17      ON  H72B   :  NB  OF  IO  REQUESTS=           21
4-7       ON  H72B   :  NB  OF  IO  REQUESTS=            6
5-7       ON  H72B   :  NB  OF  IO  REQUESTS=            3
6-7       ON  H72B   :  NB  OF  IO  REQUESTS=            3
OCS101    ON  H72B   :  NB  OF  IO  REQUESTS=            2
3-7       ON  H72B   :  NB  OF  IO  REQUESTS=           12
3-16      ON  H72B   :  NB  OF  IO  REQUESTS=            1
3-21      ON  H72B   :  NB  OF  IO  REQUESTS=           10
EA310003  ON  S72B   :  NB  OF  IO  REQUESTS=           16
H_CONF2   ON  H72B   :  NB  OF  IO  REQUESTS=            6
H_SLIB2   ON  H72B   :  NB  OF  IO  REQUESTS=            2
H_BLIB2   ON  H72B   :  NB  OF  IO  REQUESTS=          768
3-18      ON  H72B   :  NB  OF  IO  REQUESTS=           21
4-8       ON  H72B   :  NB  OF  IO  REQUESTS=            6
5-8       ON  H72B   :  NB  OF  IO  REQUESTS=            3
6-8       ON  H72B   :  NB  OF  IO  REQUESTS=            3
3-8       ON  H72B   :  NB  OF  IO  REQUESTS=            9
3-22      ON  H72B   :  NB  OF  IO  REQUESTS=           10
EA510002  ON  H72B   :  NB  OF  IO  REQUESTS=            6
STDOUT    ON  S72B   :  NB  OF  IO  REQUESTS=            4
STDERR    ON  S72B   :  NB  OF  IO  REQUESTS=            4
CPU         0.194                       PROG MISSING PAGES        0   STACKOV       0
ELAPSED     4.097                       SYS  MISSING PAGES        0
LINES           0   LIMIT     NOLIM     BACKING STORE             0   LOCKED  3133440
CARDS           0   LIMIT     NOLIM     BUFFER SIZE         2895872   CPSIZE     4096
09:22:32  STEP    COMPLETED             (MAY 07, 1996)

>>> XUFAS STEP STATISTICS         STEP=H_NCCD7
    POOLSIZE   =  4080640    USED SIZE  =    65536    NBPOOLS    =        1
    AVAIL CI   =       41    FREE CI    =      116    TOTAL CI   =      129
    SEGCR      =       17    SEGDL      =       17    ADDPATHCR  =   17        ADDPATHDL  = 17
    READIOCT   =       19    WRITEIOCT  =       45    USEDAP     =        4
    AP REQUESTS=      152    AP HITS    =      135    AP SWITCHS =        0    EXTRA APs  =  0

START    09:18:12 (MAY 07, 1996)    LINES         0
STOP     09:22:34 (MAY 07, 1996)    CARDS         0
CPU         0.194
ELAPSE      4.366
09:22:34  RESULT   JOB COMPLETED
```

B.1.2 Notes on OCS JOR

In the discussion below, a precise message time is given to help you locate the message in the JOR. Otherwise, this time is not significant to the discussion.

Only the NCC1 server is supporting IPS.

Starting Messages for Server NCC1

The second "Buffer Pool Creation" (time = 09:18:35 with BUFFER UNIT SIZE = 256 etc.) is that used for ISO, DSA, and PPAM on NCC1. This buffer size can be configured (using BUFSIZE, BUFNB of the SVR directive).

The LTH messages at time 09:18:36 and time 09:18:38 indicate the activation of the administrative LT (LT91).

The LTH messages at time 09:18:47 and time 09:18:48 indicate that the PPA has received the attentions.

The two LTH messages at time 09:18:49 indicate the activation of the PPA trace LT (LT35).

The two LTH messages at time 09:18:58 indicate the activation of the IPSS LT (LT33).

The two LTH messages at time 09:19:02 indicate the activation of the SNMP LT (LT34).

Termination Messages for Server NCC1

The "Buffer Pool Termination" (time = 09:22:10 with NUMBER OF BUFFER UNIT = 2048 etc.) is that for NCC1.

The message "TASK PPAS PRID=01 ABORTED BY SYSTEM" refers to the termination of the PPAS process for NCC1.

The message "TASK IPSS PRID=05 ABORTED BY SYSTEM" refers to the termination of the IPSS process for NCC1.

The message "TASK ADMA PRID=06 ABORTED BY SYSTEM" refers to the termination of the DSAC administration process.

The message "TASK CONF PRID=07 ABORTED BY SYSTEM" refers to the termination of the CONF process for NCC1.

The message "TASK ITRC PRID=0B ABORTED BY SYSTEM" refers to the termination of the ISO Trace process for NCC1.

The message "TASK LIPS PRID=0F ABORTED BY SYSTEM" refers to the termination of the IPS administration process.

The message "TASK SGMR PRID=10 ABORTED BY SYSTEM" refers to the termination of the SNMP manager process.

The message "TASK PTRC PRID=11 ABORTED BY SYSTEM" refers to the termination of the PPA Trace process for NCC1.

The message "TASK SNMP PRID=15 ABORTED BY SYSTEM" refers to the termination of the SNMP process for NCC1.

The message "TASK STRC PRID=19 ABORTED BY SYSTEM" refers to the termination of the IPSS trace process for NCC1.

Starting Messages for Server NCC2

The second "Buffer Pool Creation" (time = 09:19:33 with BUFFER UNIT SIZE = 256 etc.) is that for NCC2.

The LTH messages at time 09:19:34 and time 09:19:36 indicate the activation of the administrative LT (LT51).

The two LTH messages at time 09:19:47 indicate the activation of the PPA trace LT (LT55).

The LTH messages at time 09:19:48 and time 09:19:52 indicate the activation of the ISO LT (LT52).

Termination Messages for Server NCC2

The "Buffer Pool Termination" (time = 09:39:13 with NUMBER OF BUFFER UNIT = 4096 etc.) is that for NCC2.

The message "TASK PPAS PRID=02 ABORTED BY SYSTEM" refers to the termination of the PPAS process for NCC2.

The message "TASK CONF PRID=08 ABORTED BY SYSTEM" refers to the termination of the CONF process for NCC2.

The message "TASK ITRC PRID=0C ABORTED BY SYSTEM" refers to the termination of the ISO Trace process for NCC2.

The message "TASK PTRC PRID=12 ABORTED BY SYSTEM" refers to the termination of the PPA Trace process for NCC2.

The message "TASK SNMP PRID=16 ABORTED BY SYSTEM" refers to the the termination of the SNMP process for NCC2.

B.2 TNS Job OccurRence Report

EXAMPLE 1:

```
JOBID=TNS USER=OPERATOR PROJECT=OPERATOR BILLING=INSTALL RON=X593

-----------------------------------------------------------------

10:50:25             JOB EXECUTION LISTING          FEB 09, 1992

-----------------------------------------------------------------

         STEP 1
         LOAD MODULE = H_TNS (16:52 JAN 11, 1988)
         LIBRARY = SYS.HLMLIB
10:50:37 STEP STARTED XPRTY=0

         ********************************************************
         **              X B F  S T A T I S T I C S           **
         ********************************************************

          ---> BUFFER POOL CREATION
         10:50:37
         BUFFER UNIT SIZE                        :       128
         MAXIMUM NUMBER OF BUFFER UNIT           :      2048
         MINIMUM NUMBER OF BUFFER UNIT           :       512
         CURRENT NUMBER OF SEGMENT BUFFER        :         1
         TASK ASAM J=05 P=01 COMPLETED
         TASK ADM J=05 P=02 COMPLETED
         TASK GATE J=05 P=03 COMPLETED

          ---> BUFFER POOL TERMINATION
         11:16:21
         NUMBER OF BUFFER UNIT                   :       512
         NUMBER OF BUFFER UNIT USED              :        72
         EXTENSION THRESHOLD                     :       409
         CURRENT NUMBER OF SEGMENT BUFFER        :         1
           MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :        87

         ********************************************************
         **     E N D   O F   X B F   S T A T I S T I C S     **
         ********************************************************

         TASK MAIN J=05 P=00 COMPLETED
         SYSBKST ON V3B17B : NB OF IO CONNECTS=    71
         H_HUBG  ON V13B17B : NB OF IO CONNECTS=   20
         CPU      0.111     PROG MISSING SEGTS  45   STACKOV  479
         ELAPSED 25.745     SYS  MISSING SEGTS   2
         LINES   0 LIMIT NOLIM BACKING STORE 1015872 LOCKED 50624
```

```
         CARDS   0 LIMIT NOLIM BUFFER SIZE     29864 CPSIZE  2128
11:16:22 STEP COMPLETED

         START   10:50:25  LINES   0
         STOP    11:16:22  CARDS   0
         CPU        0.111
         ELAPSE    25.947
11:16:22 RESULT  JOB COMPLETED
```

EXAMPLE 2:

```
JOBID=TNS USER=OPERATOR PROJECT=OPERATOR BILLING=INSTALL RON=X998

-----------------------------------------------------------------

10:21:51            JOB EXECUTION LISTING           DEC 05, 1990

-----------------------------------------------------------------
         STEP 1
         LOAD MODULE = H_TNS (18:18 DEC 03, 1990)
         LIBRARY = SYS.HLMLIB
10:22:02 STEP STARTED XPRTY=0

         ********************************************************
         **              X B F  S T A T I S T I C S           **
         ********************************************************

          ---> BUFFER POOL CREATION
         10:22:05
         BUFFER UNIT SIZE                             :       128
         MAXIMUM NUMBER OF BUFFER UNIT                :     15872
         MINIMUM NUMBER OF BUFFER UNIT                :     15360
         CURRENT NUMBER OF SEGMENT BUFFER             :        30

          ---> BUFFER POOL EXTENSION
         10:59:37
         CURRENT NUMBER OF SEGMENT BUFFER             :        31
         NUMBER OF BUFFER UNIT                        :     15872
         NUMBER OF BUFFER UNIT USED                   :     12390
         EXTENSION THRESHOLD                          :     12697
         TASK ADM PGID=000A PRID=02 COMPLETED
         TASK ASAM PGID=000A PRID=01 COMPLETED
         TASK TRC PGID=000A PRID=03 COMPLETED

          ---> BUFFER POOL TERMINATION
         11:12:19
         NUMBER OF BUFFER UNIT                        :     15872
         NUMBER OF BUFFER UNIT USED                   :     14910
         EXTENSION THRESHOLD                          :     12697
         CURRENT NUMBER OF SEGMENT BUFFER             :        31
```

```
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       508
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       505
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       505
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       489
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       496
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       503
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       482
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       505
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       490
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       504
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       499
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       504
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       508
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       506
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       496
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       505
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       505
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       503
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       500
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       501
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       503
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       501
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       503
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       498
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       495
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       500
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       506
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       491
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       505
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       505
  MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       490

********************************************************
**     E N D   O F   X B F   S T A T I S T I C S     **
********************************************************

TASK MAIN PGID=000A PRID=00 COMPLETED
SYSBKST ON V3B17B : NB OF IO CONNECTS=   571
H_HUBG  ON V13B17B : NB OF IO CONNECTS=  312
CPU      7.342     PROG MISSING SEGTS  45   STACKOV  479
ELAPSED 50.187     SYS  MISSING SEGTS  12
LINES   0 LIMIT NOLIM BACKING STORE 1015872 LOCKED 50624
CARDS   0 LIMIT NOLIM BUFFER SIZE     29864 CPSIZE  2128
11:12:20 STEP COMPLETED

         START   10:22:02  LINES   0
         STOP    11:12:20  CARDS   0
         CPU        7.342
         ELAPSE    50.187
11:12:20 RESULT  JOB COMPLETED
```

❑

B.3 FEPS Job OccurRence Report

```
JOBID=FEPS USER=OPERATOR PROJECT=OPERATOR BILLING=INSTALL RON=X60

-----------------------------------------------------------------

11:18:20            JOB EXECUTION LISTING          FEB 09, 1988

-----------------------------------------------------------------
         STEP 1
         LOAD MODULE = H_FEPS (17:01 JAN 11, 1988 )PREINITIALIZED
         LIBRARY = SYS.HLMLIB
11:18:20 STEP STARTED XPRTY=0
         >>> FEB 09, 1988 FRONT-END  10.00


         ********************************************************
         **               X B F  S T A T I S T I C S          **
         ********************************************************

          ---> BUFFER POOL CREATION
         11:18:22

         BUFFER UNIT SIZE                          :       128
         MAXIMUM NUMBER OF BUFFER UNIT             :      2048
         MINIMUM NUMBER OF BUFFER UNIT             :       512
         CURRENT NUMBER OF SEGMENT BUFFER          :         1
         TASK ASAM J=05 P=01 COMPLETED
         TASK ADM J=05 P=02 COMPLETED

          ---> BUFFER POOL TERMINATION
         11:20:06
         NUMBER OF BUFFER UNIT                     :       512
         NUMBER OF BUFFER UNIT USED                :         0
         EXTENSION THRESHOLD                       :       409
         CURRENT NUMBER OF SEGMENT BUFFER          :         1
           MAXIMUM BUFFER UNIT USED IN DELETED BUFFER :       16

         ********************************************************
         **     E N D  O F  X B F  S T A T I S T I C S        **
         ********************************************************
```

```
*******************************************************
***                  FEPS STATISTICS               ***
*******************************************************

NB OF INWARD CONNECTION ATTEMPTS  :         0
NB OF INWARD CONNECTIONS          :         0
NB OF OUTWARD CONNECTION ATTEMPTS :         1
NB OF OUTWARD CONNECTIONS         :         1
NB OF LETTERS SENT                :         4
NB OF LETTERS RECEIVED            :         3
NB OF INWARD DISC. ATTEMPTS       :         0
NB OF INWARD DISCONNECTIONS       :         0
NB OF OUTWARD DISC. ATTEMPTS      :         1
NB OF OUTWARD DISCONNECTIONS      :         1
NB OF SELF DISCONNECTIONS         :         0
NB OF I/O ERRORS RECEIVED         :         0

*******************************************************
***                END OF FEPS STATISTICS          ***
*******************************************************
<<<
TASK MAIN J=05 P=00 COMPLETED
SYSBKST ON V3B17B : NB OF IO CONNECTS=    19
CPU      0.008     PROG MISSING SEGTS  14   STACKOV   36
ELAPSED  1.772     SYS  MISSING SEGTS   1
LINES   0 LIMIT NOLIM BACKING STORE     0   LOCKED 29808
CARDS   0 LIMIT NOLIM BUFFER SIZE     10584 CPSIZE  4240


11:20:07 STEP COMPLETED

         START   11:18:19  LINES   0
         STOP    11:20:07  CARDS   0
         CPU        0.008
         ELAPSE     1.804

11:20:07 RESULT  JOB COMPLETED
```

EXAMPLE of JOR common to TNS & FEPS

```
JOBID=TNS or FEPS      USER=OPERATOR            PROJECT=OPERATOR

-----------------------------------------------------------------
16:36:45                                     JOB EXECUTION LISTING
-----------------------------------------------------------------
          STEP 1
          LOAD MODULE = H_TNS or H_FEPS
          LIBRARY = SYS.HLMLIB
16:36:46  STEP STARTED XPRTY=0



          *******************************************************
          **           X B F     S T A T I S T I C S          **
          *******************************************************

 ---         ---> BUFFER POOL CREATION
|   |     16:36:47
|(A)|     BUFFER UNIT SIZE                              :    128
|   |     MAXIMUM NUMBER OF BUFFER UNIT                 :   2048
|   |     MINIMUM NUMBER OF BUFFER UNIT                 :    512
 ---      CURRENT NUMBER OF SEGMENT BUFFER              :      1
           .
           .
           .
 ---       ---> BUFFER POOL EXTENSION
|   |     17:55:48
|   |     CURRENT NUMBER OF SEGMENT BUFFER              :      2
|(B)|     NUMBER OF BUFFER UNIT                         :   1024
|   |     NUMBER OF BUFFER UNIT USED                    :    410
|   |     EXTENSION THRESHOLD                           :    819
 ---           .
           .
           .
          TASK ASAM J=02 P=01 COMPLETED
          TASK ADM J=02 P=02 COMPLETED
 ---       ---> BUFFER POOL TERMINATION
|   |     09:16:33
|   |     NUMBER OF BUFFER UNIT                         :   1024
|   |     NUMBER OF BUFFER UNIT USED                    :    132
|(D)|     EXTENSION THRESHOLD                           :    819
|   |     CURRENT NUMBER OF SEGMENT BUFFER              :      2
|   |
|   |       MAXIMUM BUFFER UNIT USED IN DELETED BUFFER  :    512
 ---        MAXIMUM BUFFER UNIT USED IN DELETED BUFFER  :    511
          *******************************************************
          **      E N D   O F   X B F   S T A T I S T I C S   **
          *******************************************************
```

Note: FEPS statistics appear immediately after XBF statistics.

```
---------------------------------------------------------------
          TASK MAIN J=02 P=00 COMPLETED
          SYSBKST    ON  K224   : NB OF IO CONNECTS=     26
          PRMBKST2   ON  N131   : NB OF IO CONNECTS=     50
          TMPBKST    ON  N131   : NB OF IO CONNECTS=     17
          H<-HUBG    ON  K224   : NB OF IO CONNECTS=     40
          CPU        34.666               PROG MISSING SEGTS   40
          ELAPSED   999.799               SYS MISSING SEGTS     5
          LINES           0   LIMIT NOLIM BACKING STORE    195840
          CARDS           0   LIMIT NOLIM BUFFER SIZE       43320

19:16:34  STEP COMPLETED
          START     16:36:45  LINES       0
          STOP      09:16:34  CARDS       0
          CPU         34.666
          ELAPSE     999.824
19:16:34  RESULT    JOB COMPLETED
```

❑

B.4 Explanation of JOR Listing for TNS, FEPS, and OCS

The explanation of each of the Sections follows on from the JOR listings of TNS, FEPS, and OCS.

Each field is shown with the algorithm which relates it to other values.

The following DTSVR command describes the configuration parameters of the SERVER (bolded parameter values are also handled in the JOR):

```
BP11:DTSVR TNS CODE=LONG
     GASV 14:17:30
     --> SV:TNS STATE:USED TYPE:TNS TRACE=0   NAUTO:1 TIMER:0
         ABORT:0 REPEAT:3 DEBUG:0  BUFSZ:128  BUF_MIN:512
         BUF_MAX:2048  EXTENS:80  NBCNCT:20
     COMMAND COMPLETED. 1 OBJECT SELECTED
```

B.4.1 Buffer Pool Creation

```
   ---> BUFFER POOL CREATION

   16:36:47
   ---------> time of creation

   BUFFER UNIT SIZE                                     :     128
   ----------------                                           ---
   |                                                            |
   |                              bufsz = 128 bytes per unit <--
   |
   | A segment can contain nbunitseg = 64Kbytes/bufsz = 512 units.
   |
    --> bufsz: size in bytes of each buffer unit.
        nbunitseg: number of units in a segment.
```

```
MAXIMUM NUMBER OF BUFFER UNIT                        :    2048
-----------------------------                             ----
|                                                            |
|                                    buf_max = 2048 units <--
|
| Buffer pool maximum size = bufsz * buf_max
|                          = 128   * 2048    = 262144 bytes
| i.e 4 segments of 64 Kbytes.
|
 --> buf_max : maximum number of units.
     bufpool_maxsize : Buffer pool maximum size in bytes.

MINIMUM NUMBER OF BUFFER UNIT                        :     512
-----------------------------                              ---
|                                                            |
|                                     buf_min = 512 units <--
|
| Buffer pool minimum size = bufsz * buf_min
|                          = 128   * 512    = 65536 bytes
| i.e 1 segment of 64 Kbytes
|
 --> buf_min : minimum number of units.
     bufpool_minsize : Buffer pool minimum size in bytes.
CURRENT NUMBER OF SEGMENT BUFFER                     :       1
--------------------------------                            --
|                                                            |
|                                               1 segment <--
|
 --> sgmtnb : number of segments to contain the buffer
```

B.4.2 Buffer Pool Extension

```
---> BUFFER POOL EXTENSION

17:55:48
---------> time extension was created

CURRENT NUMBER OF SEGMENT BUFFER                     :       2
--------------------------------                            --
|                                                            |
|                                  sgmtnb = 2 segments <--
|
 --> sgmtnb : number of segments

     As the buffer pool threshold is reached, XBF extends the
     buffer pool by allocating additional segments to it.

NUMBER OF BUFFER UNIT                                :    1024
---------------------                                      ---
|                                                            |
|                                  buf_nb = 1024 units <--
|    buf_nb = sgmtnb * nbunitseg
|           = 2 segments of 512 units = 1024 units
|
 --> buf_nb : current number of buffer units

NUMBER OF BUFFER UNIT USED                           :     410
--------------------------                                 ---
|                                                            |
|                                buf_nbuse = 410 units <--
|
 --> buf_nbuse : current number of buffer units used

EXTENSION THRESHOLD                                  :     819
-------------------                                        ---
|                                                            |
|                                   exthld = 819 units <--
|    exthld = buf_nb * extens
|           = 1024 * 80% = 819 units
|
 --> exthld : number of used units which must be reached to
              extend the buffer pool.
```

B.4.3 Buffer Pool Termination

```
---> BUFFER POOL TERMINATION

09:16:33
---------> time of termination


NUMBER OF BUFFER UNIT                                  :    1024

---------------------                                       ----
|                                                              |
|                                     buf_nb = 1024 units <--
|
 --> buf_nb = sgmtnb * nbunitseg


NUMBER OF BUFFER UNIT USED                             :     132
--------------------------                                   ---
|                                                              |
|                                  buf_nbuse  = 132 units <--
|
 --> buf_nbuse


EXTENSION THRESHOLD                                    :     819
-------------------                                         ----
|                                                              |
|                                      exthld = 819 units <--
|
 --> exthld = buf_nb * extens


CURRENT NUMBER OF SEGMENT BUFFER                       :       2
--------------------------------                              --
|                                                              |
|                                     sgmtnb = 2 segments <--
|
 --> sgmtnb
```

```
MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :     512
-----------------------------------------                  ---
|                                                            |
|                                     mxbfudl = 512 units <--
|
 --> mxbfudl (of first segment)


MAXIMUM BUFFER UNIT USED IN DELETED BUFFER           :     511
-----------------------------------------
|
 --> mxbfudl (of last segment)
```

B.5 List of Acronyms and Algorithms

The following list gives the acronyms used in the previous *Explanation of the JOR listing* in alphabetical order.

They are intended as a quick reference to the algorithms used for calculating the values shown against the fields.

```
buf_nb          NUMBER OF BUFFER UNIT
------          B, C and D
 unit : units    ------------------------------
                |  buf_nb = sgmtnb * nbunitseg  |
                 ------------------------------

buf_nbuse       NUMBER OF BUFFER UNIT USED
---------       B, C and D
 unit : units   number of buffer units used at time of report
                 --------------------
                |  buf_nbuse <= buf_nb|
                 --------------------

buf_max         maximum number of buffer units allocated
--------        to the buffer pool.
  unit : units  declared in SVR directive (BUFNB parameter).

buf_min         minimum number of buffer units allocated
-------         to the buffer pool.
                declared in SVR directive (BUFNB parameter).
 unit : units

bufpool_maxsize BUFFER POOL MAXIMUM SIZE
--------------  A
 unit : bytes    ----------------------------------
                |  bufpool_maxsize = bufsz * buf_max|
                |  bufpool_maxsize <= 2031616 bytes |
                 ----------------------------------

bufpool_minsize BUFFER POOL MINIMUM SIZE
--------------  A
 unit : bytes    ----------------------------------
                |  bufpool_minsize = bufsz * buf_min|
                 ----------------------------------

bufsz           BUFFER UNIT SIZE
-----           A
 unit : bytes   size of a buffer unit declared in SVR
                directive (BUFSIZE parameter).
                a segment is a multiple of bufsz.
```

```
extens          EXTENSION THRESHOLD
------          B, C and D
 unit : scalar  % declared by EXT parameter of SVR directive.
                The default value is 80%

exthld          UNIT EXTENTION THRESHOLD
------          B, C and D
 unit : units    ---------------------------
                |  exthld = buf_nb * extens |
                 ---------------------------
                number of used units which must be reached to
                extend the buffer pool.

mxbfudl         MAXIMUM BUFFER UNITS USED IN DELETED BUFFER
-------         C and D
 unit : units   number of units that have been used in the
                segment that has just been deleted.
                 --------------------
                |  mxbfudl <= bufsz  |
                 --------------------

nbunitseg       NUMBER OF UNITS IN A SEGMENT
---------       A, B, C and D
 unit : units    -----------------------------
                |  nbunitseg = 64Kbytes/bufsz |
                 -----------------------------

sgmtnb          CURRENT NUMBER OF SEGMENT BUFFER
------          A, B, C and D
 unit:segments   --------------------------
                |  sgmtnb =>  1 (minimum)  |
                |  sgmtnb <= 31 (maximum)  |
                 --------------------------
```

B.6 QMON Jor Listing

```
JOBID=QMON USER=OPERATOR PROJECT=OPERATOR BILLING=INSTALL RON=X67

-----------------------------------------------------------------

11:19:02             JOB EXECUTION LISTING           FEB 09, 1988

-----------------------------------------------------------------

         STEP 1
         LOAD MODULE = H_QMON (16:31 DEC 28, 1987)
         LIBRARY = SYS.HLMLIB
11:19:06 STEP STARTED XPRTY=0


         ********************************************************
         *******************  QMON STATISTICS  *****************
         ********************************************************

         BMAM.      EXTAB SEG SIZE        : 1158 DB SIZE      :
                                                 ----------------
                                                                |
                    declared by QFBLKSZ of NETWORK directive <--


         BMAM.      DBUF POOL SEG SIZE    :12424
         BMAM.      CBUF = POOL SEG SIZE  :38000 CB SIZE      :76
                                                 ----------------
                                                                |
                    declared by QMBLKSZ of NETWORK directive <--


         QMON.      QMTAB SEG SIZE        : 1947
         QMON.      SP WORK AREAS SIZE    : 7326
                    ----------------------------
                    |
                     -> size of stream processor for handling
                        mode conversion : NL, UN and MK


         BMAM.      PEAK NUMBER OF CB USED:   14
                    ----------------------------
                    |
                     -> maximum number of memory blocks used
```

```
BMAM.      PEAK NUMBER OF DB USED:    3
           ----------------------------
           |
            -> maximum number of disk blocks used


BMAM USERS.PEAK NB OF DB USED    :    3
           ----------------------------
           |
            -> maximum disk blocks used by applications


QMON.      PEAK NB OF DB USED    :    3
           ----------------------------
           |
            -> max number of disk blocks used by QMON
QMON.      PEAK NB OF QMGE USED  :    1 QMGETAB SIZE : 8
           ---------------------------- ----------------
           |                                           |
            -> maximum number of connections between   |
               QMON and VCAM                           |
                                                       |
                         size of message group table <-


QMON.      PEAK NB OF CICD USED  :    1 CICD TAB SIZE:12
           ---------------------------- ----------------
           |                                           |
            -> max number of input CDs used by QMON    |
                                                       |
                              size of input CD table <-


QMON.      PEAK NB OF COCD USED  :    0 COCD TAB SIZE:48
           ---------------------------- ----------------
           |                                           |
            -> max number of output CDs used by QMON   |
                                                       |
                             size of output CD table <-


QMON.      PEAK LENGTH OF RESWQ  :    0 RESWQTAB     :10
           ---------------------------- ----------------
           |                                           |
            -> max length of resource waiting queue    |
               used by QMON                            |
                                                       |
                      size of resource waiting queue <-
```

```
         ********************************************************
         *****************  END OF STATISTICS  ******************
         ********************************************************

         TASK MAIN J=06 P=00 COMPLETED
         FP07.IFN:H_QC_FMS HAS BEEN CLOSED BY SYSTEM
         SYSBKST  ON V3B17B : NB OF IO CONNECTS=    43
         H_QC_FMS ON V13B17B : NB OF IO CONNECTS=    0
         CPU      0.004     PROG MISSING SEGTS  35   STACKOV   13
         ELAPSED  1.017     SYS  LISSING SEGTS   2
         LINES   0 LIMIT NOLIM BACKING STORE  195360 LOCKED 10000
         CARDS   0 LIMIT NOLIM BUFFER SIZE      4696 CPSIZE  2272

11:20:07 STEP COMPLETED
         START   11:19:02  LINES   0
         STOP    11:20:07  CARDS   0
         CPU        0.004
         ELAPSE     1.094

11:20:07 RESULT  JOB COMPLETED
```

B.7 RAEH Job Occurrence Report

In the Job Occurrence Report of the RAEH server the following lines supply statistical information on the administrative objects handled by the server components *NAD* and *MUX*:

- Messages in GCOS format
- Records in AEP format
- Unsollicited messages
- Commands and responses
- and Trace levels.

```
STATISTICS FROM NADMUX (NUMBER OF RECORDS) :
--------------------------------------------
     RECEIVED  AEP   COMMANDS     :      nnnn
     SENT      AEP   COMMANDS     :      nnnn
     RECEIVED  GCOS  MESSAGES     :      nnnn
     SENT      GCOS  MESSAGES     :      nnnn
     RECEIVED  AEP   RESPONSES    :      nnnn
     SENT      AEP   RESPONSES    :      nnnn
     SENT      UMS                :      nnnn
     UMS RECEIVED AND SENT TO LOG :      nnnn
     UMS RECEIVED AND PURGED      :      nnnn
     COMMANDS REJECTED BY NADMUX  :      nnnn

TRACE LEVEL: NAD = nnnn  LOG = nnnn  NER = nnnn  SBMON = nnnn
```

❑

C. Handling the Site Terminal Table

THE INFORMATION GIVEN IN THE SECTION ONLY PERTAINS TO SITES WHICH HAVE SPECIAL TERMINAL REQUIREMENTS. RECONFIGURING THE TERMINAL TABLES IS TO BE DONE WITH THE ADVICE AND UNDER THE GUIDANCE OF BULL INSTALLATION PERSONNEL.

C.1 Site Terminal Table

The Site Terminal Table is a binary table which contains the complete description of all the terminal models available on the site. It gives all the characteristics for each terminal such as:

- cursor positioning
- and the class-of-protocol used by the terminal.

The information in this table is used by the presentation and session layers. The presentation layer is concerned by the type of terminal support such as formatted or graphic and by the type of text terminal adaptor.

C.2 MAINTAIN_TERMINAL Utility (MNTERM)

The MNTERM utility (MAINTAIN_TERMINAL) builds, accesses and initializes the Site Terminal Table.

Standard Terminal Support

MNTERM initializes the Site Terminal Table by compiling all the standard terminal source descriptions contained in the SYS.HSLLIB..H_TERM member.

The standard terminal source descriptions must never be altered.

Non-Standard Terminal Support

To define new terminals specific to the the site or to modify the description of standard terminals:

- either supply MNTERM the additional terminal descriptions
- or after the compilation of the standard terminals, alter the characteristics of some of them directly in the Site Terminal Table.

C.3 Terminal Configuration Directives

The low-level terminal configuration directives which address MNTERM are:

- **DEFTERM** for defining a new terminal type by completely listing its characteristics.
- **DEFLIKE** for defining a new terminal type by specifying only the differences between it and one *already generated* whether standard or non-standard. DEFLIKE must therefore appear after the DEFTERM which it affects. DEFLIKE can also refer to another DEFLIKE.
- **UPTERM** for modifying the characteristics of a terminal type *already generated* whether standard or non-standard. UPTERM must appear after either the DEFTERM or the DEFLIKE which it affects.

The term *already generated* means that the directive describing the terminal type appears immediately before the directive in question. DEFTERM + DEFLIKE + UPTERM denotes the sequence that each of the directives has on the one preceding it, each subsequent directive considering the one before it *already generated*.

C.4 Initializing the Site Terminal Table

The Site Terminal Table is created and initialized through the CRNETGEN command which invokes in sequence the NETGEN utility, then MNTERM.

If only standard terminal models are used, there is nothing to specify, because NETGEN knows when the terminal generation is needed and where to find the source descriptions.

However, if terminals other than the standard models are to be configured, the Site Terminal Table is initialized at basic network configuration generation by the CRTERM parameter of the CRNETGEN command. The CRTERM parameter specifies where to retrieve the descriptions of private non-standard terminals. See Section 2.

C.5 Creating the Terminal Configuration

In addition to generating network configurations, NETGEN also generates the terminal configuration. The terminal configuration is a table of known terminal models at the site and is:

- created if SAVE and/or ENABLE are requested
- and forced if CRTERM=STANDARD is specified in the CRNETGEN command.

Each time the terminal configuration is generated, its directives processed are listed in the NETGEN execution report.

A set of standard directives is delivered in the SYS.HSLLIB..H_TERM subfile which relieves the user of the task of terminal configuration. If needed, however, the user can generate additional terminal directives in a private library member by specifying CRTERM=member.

Like network configurations, the terminal configuration:

- can be stored in either a temporary, permanent or running workspace depending on the parameters of the CRNETGEN command
- and handled by the LOAD and SAVE functions of NETGEN.

When generating terminal configurations for several remote systems on a dedicated DPS 7000, the standard terminal configuration used is that of the DPS 7000 on which NETGEN is run. This implies full compatibility in terminal configurations among all the remotely generated DPS 7000s. If this is not the case, the terminal configuration can be adapted for each targeted site by specifying CRTERM=member.

C.6 DFL (DEFLIKE) Directive

THE DEFLIKE DIRECTIVE SHOULD ONLY BE USED WITH THE ADVICE AND UNDER THE GUIDANCE OF BULL INSTALLATION PERSONNEL.

Purpose

This directive applies only to the terminal configuration description. It allows defining a new terminal type using an already existing one (previously declared by a DFT or by another DFL directive) and specifying only the attribute differences between the targeted type and its specified model.

The syntax is almost that of the DFT directive with the exception that:

- the only mandatory parameters are: EXTYPE, DSAMODEL and LIKE
- ***the default value for each parameter is that actually retrieved in the model.***

Syntax

```
{ DFL     }                                        ,DSAMODEL=hexa4
{         }             EXTYPE=name12
{ DEFLIKE }                                        ,LIKE=name12

[,EMULATE=hexa4 ]               [,INTYPE=hexa2 ]

[,COMP_INTYPE=hexa2 ]           [,COMFEAT={ PAD | CSX25 | NONE } ]

[,ADDMODE={ VIP | VIP7800 } ]

,SUBTYPES=( [CPU] [DSK] [CAS] [PRT] [CRT] [KCT] [KPR] [KB] ) (1)

[         { TTY   } ]           [         { TTY   } ]
[         { VIP   } ]           [         { VIP   } ]
[,PROC1 = { BSC   } ]           [,PROC2 = { BSC   } ]
[         { TCV   } ]           [         { TCV   } ]
[         { HDLC  } ]           [         { HDLC  } ]
[         { ANY   } ]           [         { TTY_R } ]

[,SYSHEAD=hexa24 ]              [,SYSTRAIL=hexa24 ]

[,DELCHAR=hexa2 ]               [,INBFSZ=dec5 ]
```

```
[ { PL | PAGELG }=dec3 ]          [ { PMX | PAGEMX }=dec3 ]

[,{ LL | LINELG }=dec3 ]          [,{ LMX | LINEMX }=dec3 ]

[,FEATURES=( [ROLL] [WRAP] [AUTOLF] [AUTOFOLD] [HTAB]   (1)
             [VTAB] [APL] [BS] [LCIN] [LCOUT] [NONE] ) ]

[,MESSLG=dec5 ]                   [,PRESCLASS=dec1 ]

[,TEXTTYPE=dec2 ]                 [,CSETTYPE=dec1 ]
[,PSWHIDE=hexa24 ]                [,PSWCLEAR=hexa24 ]

[,PAGECLEAR=hexa8 ]               [,FORMSTYPE=dec2 ]

[,GKSTYPE=dec1 ]                  [,TABCLEAR=hexa8 ]

[,TABSET=hexa24 ]                 [,TABMODE={ SDP | NSDP } ]

{ FRW | FORWARD }=hexa8 ]         [,TOSTATUS=hexa24 ]

[,OUTSTATUS=hexa24 ]              [,ADDRHEADER=hexa8 ]

[,ADDRTRAILER=hexa8 ]             [,ADDRMODE=dec1 ]

[,PLHEADER=hexa16 ]               [,PLTRAILER=hexa16 ]

[,PLMODE=dec1 ]                   [,ROLLON=hexa16 ]

[,ROLLOFF=hexa16 ]                [,INITSTRING=hexa24 ]

[,BLOCK=bool ]                    [,ENQ=hexa16 ]
```

(1) This list cannot be empty and cannot start with a comma. Commas are used to separate items.

Parameter Description	
EXTYPE	*EXTYPE=name12* identifies the terminal and must be unique.
DSAMODEL	*DSAMODEL=hexa4* identifies the terminal. This code must also be unique so that a one-to-one mapping exists between it and the EXTYPE name. It must be recognized by the presentation layer of the correspondent and is a network-wide code for DSA200. The presentation layer also uses DSAMODEL to retrieve all the characteristics of this terminal: DSAMODEL=10xx: Information processors DSAMODEL=20xx: Bull standard terminals DSAMODEL=40xx: Other Bull terminals *DSAMODEL=60xx:* User-specific terminals.
LIKE	*LIKE=name12* identifies an already existing terminal type to be taken as a model and from which the new terminal is described.

OPTIONS

All other parameters are optional and are described in the DFT directive. The ***default*** for each option of the DFL directive is not that specified in the DFT directive syntax but that actually retrieved in the terminal description specified as model by the ***LIKE*** parameter.

C.7 DFT (DEFTERM) Directive

THE DEFTERM DIRECTIVE SHOULD ONLY BE USED WITH THE ADVICE AND UNDER THE GUIDANCE OF BULL INSTALLATION PERSONNEL.

Purpose

This directive applies only to the terminal configuration description. It defines a new terminal type not yet existing in the H_TERM standard file which lists ***all*** terminal types supported on the site by the GCOS 7 applications.

Syntax

```
{ DFT     }                                        ,DSAMODEL=hexa4
{         }          EXTYPE=name12
{ DEFTERM }                                        ,INTYPE=hexa2

,SUBTYPES=( [CPU] [DSK] [CAS] [PRT] [CRT] [KCT] [KPR] [KB] ) (1)

[,EMULATE={ 0000 | hexa4 }]      [,COMP_INTYPE={ 00 | hexa2 }]

[,COMFEAT={ NONE|PAD|CSX25 }]    [,ADDMODE={ VIP | VIP7800 }]

       { TTY  }                  [       { TTY   } ]
       { VIP  }                  [       { VIP   } ]
,PROC1={ BSC  }                  [,PROC2={ BSC   } ]
       { TCV  }                  [       { TCV   } ]
       { HDLC }                  [       { HDLC  } ]
       { ANY  }                  [       { TTY_R } ]

[,SYSHEAD={ 0D25 | hexa24 } ]    [,SYSTRAIL=hexa24 ]

[,DELCHAR={ 00 | hexa2 }]        [,INBFSZ={ 4096 | dec5 }]

[,{ PL |PAGELG }={ 24 | dec3 }]  [,{ PMX |PAGEMX }={ 24 | dec3 }]

[,{ LL |LINELG }={ 80 | dec3 }]  [,{ LMX |LINEMX }={ 80 | dec3 }]

[,FEATURES = ( [NONE] [ROLL] [WRAP] [AUTOLF] [AUTOFOLD]    (1)
               [HTAB] [VTAB] [APL] [BS] [LCIN] [LCOUT] ) ]

[,MESSLG={ 4096 | dec }]         [,PRESCLASS={ 0 | 1 | 2 | 3 }]
```

```
[,TEXTTYPE={ 0 | dec2 }]          [,CSETTYPE={ 0|1|2|3|4|5|6 }]
[,PSWHIDE=hexa24 ]                [,PSWCLEAR=hexa24 ]
[,PAGECLEAR=hexa8 ]               [,FORMSTYPE={ 0 | dec2 }]
[,GKSTYPE={ 0|1|2|3|4 }]          [,TABCLEAR=hexa8 ]
[,TABSET=hexa24 ]                 [,TABMODE={ NSDP | SDP }]
[,{ FRW | FORWARD }={ 12 | hexa8 } ]
[,TOSTATUS=hexa24 ]               [,OUTSTATUS=hexa24 ]
[,ADDRHEADER=hexa8]               [,ADDRTRAILER=hexa8]
[,ADDRMODE={ 0 | 1 | 2 | 3 }]     [,PLMODE={ 0 | 1 | 2 | 3 }]
[,PLHEADER=hexa16]                [,PLTRAILER=hexa16]
[,ROLLON=hexa16 ]                 [,ROLLOFF=hexa16 ]
[,INITSTRING=hexa24 ]             [,BLOCK={ 0 | 1 } ]
[,ENQ=hexa16 ] ;
```

(1) This list cannot be empty and cannot start with a comma. Commas are used to separate items.

Parameter Description

EXTYPE *EXTYPE=name12* identifies the terminal and must be unique.

DSAMODEL *DSAMODEL=hexa4* identifies the terminal. This code must be unique for one-to-one mapping with EXTYPE. It must be recognized by the presentation layer of the correspondent and is the network-wide code for DSA200.

The presentation layer uses DSAMODEL to retrieve all the terminal characteristics:
DSAMODEL=10xx: Information processors
DSAMODEL=20xx: Bull standard terminals
DSAMODEL=40xx: Other Bull terminals
DSAMODEL=60xx: User-specific terminals.

EMULATE	*EMULATE=hexa4* supplies a DSAMODEL to be passed to the correspondent either when the terminal manager emulates or when the subset of the functionalities of the real terminal does map to a standard terminal. *EMULATE=0000:* ***Default*** *T*he value passed to the correspondent is that of DSAMODEL.
INTYPE	Local code for the protocol used by the terminal manager: *INTYPE=07:* Asynchronous printing terminal *INTYPE=0C:* Asynchronous APL terminal *INTYPE=0E:* VIP7801 family *INTYPE=0F:* Asynchronous TTY terminal *INTYPE=1E:* TELEX terminal *INTYPE=1F:* VIDEOPAD terminal *INTYPE=28:* VIP7760 family *INTYPE=2A:* PVE emulated VIP *INTYPE=2E:* VIP7800 family *INTYPE=30:* QUESTAR family phase 1 *INTYPE=40:* OLIVETTI terminals *INTYPE=50:* IBM3270 terminals *INTYPE=60:* CPU transparent mode
COMP_INTYPE	*COMP_INTYPE=hexa2* defines a subclass of terminals for the Terminal Manager as a complement to INTYPE. ***Example:*** COMP_INTYPE may identify the members of the family VIP7800 if the value of the INTYPE parameter is 2E. *COMP_INTYPE=00:* ***Default***, No subclass defined.
COMFEAT	Communications features determining the connection of the terminal type over the network. *COMFEAT=NONE:* ***Default*** *COMFEAT=PAD:* Connection to a PAD concentrator *COMFEAT=CSX25:* Connection to a CSX25 concentrator

ADDMODE	**Applicable only to VIP terminals.** Device addressing mode determines the values used in the ADDR byte to address the devices of the VIP station. *ADDMODE=VIP: **Default***, VIP mode *ADDMODE=VIP7800*: VIP 7800 mode.
PROC1	Specifies the line procedure supported by the terminal. `PROC1=ANY:` no control on the line procedure `PROC1=TTY:` TTY-like asynchronous terminals `PROC1=BSC:` Binary Synchronous Communication `PROC1=TCV:` Olivetti `PROC1=HDLC:` High-level Data Link Control `PROC1=VIP:` VIP-like synchronous terminals
PROC2	Specifies an alternate value for PROC1. No default.
SYSHEAD	*SYSHEAD=hexa24* is an EBCDIC control string used to prefix system service-messages which are directed to the terminal. *SYSHEAD=0D25: **Default***, Carriage return and line feed.
SYSTRAIL	*SYSTRAIL=hexa24* is an EBCDIC control string to be appended to System service-messages which are directed to the terminal. ***Default**:* no string defined.
DELCHAR	*DELCHAR=hexa2* is an EBCDIC internal representation of the default character to be used as erase character on TTY asynchronous terminals for text editing. *DELCHAR=00: **Default***, No erase character defined.
INBFSZ	*INBFSZ=dec5* is the input buffer size of the terminal expressed as a number of bytes. ***Default**:* 4096 bytes.
PL alias **PAGELG**	*PAGELG=dec3* is the default page length of the terminal expressed as a number of lines. *pagelg <= pagemx, **Default***: 24 lines. *PAGELG=0:* Not paged terminal.
PMX alias **PAGEMX**	*PMX=dec3* is the maximum value authorized for PAGELG. ***Default**:* 24 lines.

LL alias **LINELG**

LINELG=dec3 is the default line length of the terminal expressed as a number of characters.

linelg <= linemx, ***Default***: 80 characters.

LMX alias **LINEMX**

LMX=dec3 is the maximum value authorized for LINELG.
Default: 80 characters.

MESSLG

Maximum length of message to be transmitted by the terminal in number of bytes.
Default: 4096 bytes.

FEATURES

List of up to 10 features available on the terminal:

FEATURES=NONE:	***Default***, No feature available
FEATURES=ROLL:	Auto rolling up
FEATURES=WRAP:	Auto wrapping up
FEATURES=AUTOLF:	Auto linefeed
FEATURES=AUTOFOLD:	Auto linefold
FEATURES=HTAB:	Horizontal tabulations
FEATURES=VTAB:	Vertical tabulations
FEATURES=APL:	APL character set
FEATURES=BS:	Backspace
FEATURES=LCIN:	Lower case character in
FEATURES=LCOUT:	Lower case character out

PRESCLASS

Presentation class:
PRESCLASS=0: ***Default***
Allows only linefeed as TTY35 or TN300
PRESCLASS=1:
Linefeed and formfeed as TTU8124 or PRT1220
PRESCLASS=2:
Direct cursor address: VIP, DKU, AJ832, MINITEL, TK4105
PRESCLASS=3: as IBM3270

TEXTTYPE	It specifies the type of coding in text mode as follows: *TEXTTYPE=0:* ***Default***, Terminal with only CR-LF *TEXTTYPE=1:* TTY33, TTY35, TTY37 *TEXTTYPE=2:* TN300, TN1200 *TEXTTYPE=3:* TTU8124, TTU8126 TEXTTYPE=4: VIP7001 *TEXTTYPE=5:* VIP7700, VIP7760 *TEXTTYPE=6:* VIP7801/7804, VIP7813/7814, PC7800/7800T *TEXTTYPE=7:* DKU7005, DKU7006, DKU7007, TTX35 *TEXTTYPE=8:* DKU7001, DKU7002 *TEXTTYPE=9:* DKU7105, DKU7211, DKU7107 *TEXTTYPE=10:* AJ832, AJ833 TEXTTYPE=11: MINITEL TEXTTYPE=12: TK4105 TEXTTYPE=13: PRT1220 *TEXTTYPE=15:* IBM3270, IBM3278, IBM3279, IBM3287
CSETTYPE	Type of character set coding: *CSETTYPE=0:* ***Default***, No PLW, no CSPS *CSETTYPE=1:* PRT1220 (SS2=19) *CSETTYPE=2:* DKU7107 (SS2=27C5) *CSETTYPE=3:* MINITEL *CSETTYPE=4:* IBM3278 (PLW + APL + CSPS) *CSETTYPE=5:* VIP7800 (CSPS, no PLW) *CSETTYPE=6:* AJ832, AJ833 (APL)
PSWHIDE	*PSWHIDE=hexa24* is an EBCDIC control string which conceals input on terminal. ***Default****:* no string defined.
PSWCLEAR	*PSWCLEAR=hexa24* is an EBCDIC control string which returns the terminal from conceal-input to normal mode. ***Default****:* no string defined.
PAGECLEAR	*PAGECLEAR=hexa8* is an EBCDIC control string which clears the current page on terminal. ***Default****:* no string defined.

FORMSTYPE Specifies the type of coding in forms mode:

FORMSTYPE=0: **Default**, No forms
FORMSTYPE=1: VIP7001, VIP7700
FORMSTYPE=2: VIP7760
FORMSTYPE=3: IBM3270
FORMSTYPE=4: DKU7005, DKU7006, DKU7007, TTX35
FORMSTYPE=5: VIP7801/7801T/7804, VIP7813/7814
FORMSTYPE=6: DKU7105, DKU7107
FORMSTYPE=7: DKU7211

FORMSTYPE=8: MINITEL
FORMSTYPE=9: IBM3278, IBM3279, IBM3287
FORMSTYPE=10: PC7800

GKSTYPE Specifies the type of coding in graphics mode:

GKSTYPE=0: ***Default***, No graphics
GKSTYPE=1: AJ832
GKSTYPE=2: AJ833
GKSTYPE=3: TK4105
GKSTYPE=4: PRT1220.

TABCLEAR *TABCLEAR=hexa8* is an EBCDIC control string to clear tabulations.
Default*:* no string defined.

TABSET *TABSET=hexa8* is an EBCDIC control string to set tabulations.
Default*:* no string defined.

TABMODE Tabulations set mode.
TABMODE=NSDP: ***Default***
The TABSET string has to be repeated at each location in the line where a tabulation stop is to be set.
TABMODE=SDP: The TABSET string sets all tabulations at one go, in compliance with the SDP standard.

FRW alias **FORWARD** *FRW=hexa8* is an EBCDIC control string for forward space.
Default: 12.

TOSTATUS *TOSTATUS=hexa24* is an EBCDIC control string giving access to the status line.
Default*:* no string defined.

OUTSTATUS	*OUTSTATUS=hexa24* is an EBCDIC control string giving back access out of the status line. ***Default****:* no string defined.
ADDRHEADER	*ADDRHEADER=hexa8* is an EBCDIC control string which precedes the line and column definition in the cursor positioning string. ***Default****:* no string defined.
ADDRTRAILER	*ADDRTRAILER=hexa8* is an EBCDIC control string which follows the line and column definition in the cursor positioning string. ***Default****:* no string defined.
ADDRMODE	**Applicable only if PRESCLASS=2.** Cursor addressing mode. *ADDRMODE=0:* ***Default*** VIP-like mode: the line and column numbers are coded according to the VIP protocol. *ADDRMODE= 1 or 2 or 3:* All cursor addressing is from column 1, the line number is directly issued as a character string of 1 or 2 or 3 bytes.
PLHEADER	*PLHEADER=hexa16:* EBCDIC control string preceding the number of lines in the page-length setting command. ***Default****:* no string defined.
PLTRAILER	*PLTRAILER=hexa16:* EBCDIC control string which follows the number of lines in the page-length setting command. ***Default****:* no string defined.
PLMODE	**Applicable only if PLHEADER is specified**. Page length coding mode. *PLMODE=0:* ***Default -*** as TTU8124 *PLMODE=1:* as TK4105 *PLMODE=2 or 3:* Number of lines in decimal on 2 or 3 bytes.
ROLLON	*ROLLON=hexa16:* EBCDIC control string to set rollup mode. ***Default****:* no string defined.

ROLLOFF	*ROLLOFF=hexa16:* EBCDIC control string to reset rollup mode. ***Default****:* no string defined.
INITSTRING	*INITSTRING=hexa24:* EBCDIC control string to initialize terminal. ***Default****:* no string defined.
BLOCK	Capability to send data in block mode. *BLOCK=0:* ***Default*** Terminal does not support block mode. *BLOCK=1:* Terminal supports block mode.
ENQ	*ENQ=hexa16:* Enquiry EBCDIC control string sent for prompting a terminal to answer back its main capabilities. ***Default****:* no string defined

C.8 UPT (UPTERM) Directive

THE UPTERM DIRECTIVE SHOULD ONLY BE USED WITH THE ADVICE AND UNDER THE GUIDANCE OF BULL INSTALLATION PERSONNEL.

Purpose

This directive applies only to the terminal configuration description. It alters any parameter of a terminal description in the terminal table generated either in the H_TERM standard file or defined by the user through the DFT or DFL directive.

The syntax is that of the DFT directive with the exception that the only mandatory parameter is EXTYPE and that only the parameters specified are modified, all others omitted are left unchanged.

Parameters which are not specified retain their original values default or explicitly defined.

Syntax

```
{ UPT    }                                    [,DSAMODEL=hexa4 ]
{        }           EXTYPE=name12
{ UPTERM }                                    [,EMULATE=hexa4 ]

[,INTYPE=hexa2 ]                    [,COMP_INTYPE=hexa2 ]

[,COMFEAT={ NONE | PAD | CSX25 }]   [,ADDMODE={ VIP | VIP7800 }]

,SUBTYPES=( [CPU] [DSK] [CAS] [PRT] [CRT] [KCT] [KPR] [KB] ) (1)

[       { TTY   } ]                 [       { TTY   } ]
[       { VIP   } ]                 [       { VIP   } ]
[,PROC1={ BSC   } ]                 [,PROC2={ BSC   } ]
[       { TCV   } ]                 [       { TCV   } ]
[       { HDLC  } ]                 [       { HDLC  } ]
[       { ANY   } ]                 [       { TTY_R } ]

[,SYSHEAD=hexa24 ]                  [,SYSTRAIL=hexa24 ]

[,DELCHAR=hexa2 ]                   [,INBFSZ=dec5 ]

[,{ PL | PAGELG }=dec3 ]            [,{ PMX | PAGEMX }=dec3 ]

[,{ LL | LINELG }=dec3 ]            [,{ LMX | LINEMX }=dec3 ]
```

```
[,FEATURES=( [ROLL] [WRAP] [AUTOLF] [AUTOFOLD] [HTAB]   (1)
             [VTAB] [APL] [BS] [LCIN] [LCOUT] [NONE] ) ]

[,MESSLG=dec5 ]                    [,PRESCLASS=dec2 ]

[,TEXTTYPE=dec2 ]                  [,CSETTYPE=dec2 ]

[,PSWHIDE=hexa24 ]                 [,PSWCLEAR=hexa24 ]

[,PAGECLEAR=hexa8 ]                [,FORMSTYPE=dec2 ]

[,GKSTYPE=dec2 ]                   [,TABCLEAR=hexa8 ]

[,TABSET=hexa24 ]                  [,TABMODE={ NSDP | SDP }]

[,{ FRW | FORWARD }=hexa8 ]        [,TOSTATUS=hexa24 ]

[,OUTSTATUS=hexa24 ]               [,ADDRHEADER=hexa8 ]

[,ADDRTRAILER=hexa8 ]              [,ADDRMODE=dec2 ]

[,PLMODE=dec1 ]                    [,PLHEADER=hexa16 ]

[,PLTRAILER=hexa16 ]               [,ROLLON=hexa16 ]

[,ROLLOFF=hexa16 ]                 [,INITSTRING=hexa24 ]
```

(1) This list cannot be empty and cannot start with a comma. Commas are used to separate items.

Parameter Description

EXTYPE *EXTYPE=name12:* Identifies the terminal description to be updated. No default.

OPTIONS

All other parameters are fully optional and are described in the DFT directive. The only difference is that in the terminal description specified in the EXTYPE parameter only the values of the parameters present in the UFT directive are updated while all others remain unchanged.

C.9 Examples

The following examples show:

- a SYS.HSLLIB..H_TERM standard file from which a subset can be tailored to include only the terminals needed at the site. The member delivered to a site may be different from this example.
- a private secondary file MY_SLLIB..MY_TERM_MB which creates the terminal MY_TERM from the standard TTU8124 terminal model description using DEFLIKE and UPTERM directives.

These two subfiles are by NETGEN by simply specifying in the CRNETGEN command:

```
CRNETGEN ... CRTERM=MY_TERM_MB  SLLIB=MY_SLLIB ... ;
```

C.9.1 Standard Terminal Models: H_TERM

```
DEFTERM EXTYPE=AJ832, DSAMODEL=2015, INTYPE=0C, COMFEAT=PAD,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=2, TEXTTYPE=10, CSETTYPE=6,
PSWHIDE=27D1, PSWCLEAR=27D2, PAGECLEAR=0C070707,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=66, PAGEMX=0,
LINELG=132, LINEMX=158, MESSLG=128, FORMSTYPE=0, GKSTYPE=1,
TABCLEAR=2716, TABSET=27F1, ADDRHEADER=0D27C2, ADDRMODE=3,
PLHEADER=27C3, PLTRAILER=275E, PLMODE=3, BLOCK=0;


DEFTERM EXTYPE=AJ833, DSAMODEL=201D, INTYPE=0C, COMFEAT=PAD,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=2, TEXTTYPE=10, CSETTYPE=6,
PSWHIDE=27D1, PSWCLEAR=27D2, PAGECLEAR=0C070707,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=66, PAGEMX=00,
LINELG=158, LINEMX=148, MESSLG=6144, FORMSTYPE=0, GKSTYPE=2,
TABCLEAR=2716, TABSET=27F1, ADDRHEADER=0D27C2, ADDRMODE=3,
PLHEADER=27C3, PLTRAILER=275E, PLMODE=3, BLOCK=0;

DEFTERM EXTYPE=BTT7300, DSAMODEL=2035, INTYPE=25,
SUBTYPES=(KCT, CAS), ADDMODE=VIP, PROC1=VIP, PRESCLASS=0,
TEXTTYPE=0, CSETTYPE=0,
PAGECLEAR=0D25, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=0, PAGEMX=0, LINELG=0, LINEMX=0, MESSLG=1024,
FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;
```

```
DEFTERM EXTYPE=DKU7001, DSAMODEL=2019, INTYPE=0F,
COMFEAT=PAD, SUBTYPES=(KB, KCT, CRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL), PRESCLASS=2, TEXTTYPE=8, CSETTYPE=0,
PSWCLEAR=2779, PAGECLEAR=27C827D1, SYSHEAD=0D25,
SYSTRAIL=0D25, DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=2048, FORMSTYPE=0, GKSTYPE=0,
FORWARD=27C3, ADDRHEADER=278640, ADDRMODE=0,
INITSTRING=27C827D1, BLOCK=0;

DEFTERM EXTYPE=DKU7002, DSAMODEL=201A, INTYPE=0F,
COMFEAT=PAD, SUBTYPES=(KB, KCT, CRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL), PRESCLASS=2, TEXTTYPE=8, CSETTYPE=0,
PSWCLEAR=2779, PAGECLEAR=27C827D1, SYSHEAD=0D25,
SYSTRAIL=0D25, DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=6144, FORMSTYPE=0, GKSTYPE=0,
FORWARD=27C3, ADDRHEADER=278640, ADDRMODE=0,
INITSTRING=27C827D1, BLOCK=0;

DEFTERM EXTYPE=DKU7005, DSAMODEL=203D, INTYPE=30,
SUBTYPES=(KB, KCT, CRT, PRT), ADDMODE=VIP, PROC1=VIP,
FEATURES=(AUTOLF, HTAB), PRESCLASS=2, TEXTTYPE=7, CSETTYPE=0,
PSWHIDE=274AF894, PAGECLEAR=0C, SYSHEAD=0D25, SYSTRAIL=0D25,
INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80, LINEMX=80,
MESSLG=6144, FORMSTYPE=4, GKSTYPE=0, TABCLEAR=27F2,
TABSET=27F1, FORWARD=12, ADDRHEADER=13, ADDRMODE=0,
ADDRTRAILER=40, BLOCK=1;

DEFTERM EXTYPE=DKU7007, DSAMODEL=2040, INTYPE=30,
COMFEAT=CSX25, SUBTYPES=(KB, KCT, CRT, PRT, DSK),
ADDMODE=VIP, PROC1=VIP, FEATURES=(AUTOLF, HTAB), PRESCLASS=2,
TEXTTYPE=7, CSETTYPE=0, PSWHIDE=274AF894, PAGECLEAR=0C,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=6144, FORMSTYPE=4, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, FORWARD=12, ADDRHEADER=13,
ADDRMODE=0, ADDRTRAILER=40, BLOCK=1;

DEFLIKE EXTYPE=DKU7102,LIKE=DKU7002,DSAMODEL=201B,CSETTYPE=1;

DEFTERM EXTYPE=DKU7107, DSAMODEL=2042, INTYPE=30,
COMFEAT=CSX25, SUBTYPES=(KB, KCT, CRT, PRT, DSK),
ADDMODE=VIP, PROC1=VIP, FEATURES=(AUTOLF, HTAB), PRESCLASS=2,
TEXTTYPE=9, CSETTYPE=2, PSWHIDE=274AF894, PAGECLEAR=0C,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=6144, FORMSTYPE=6, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, FORWARD=12, TOSTATUS=274AF0A5,
OUTSTATUS=274AF1A5, ADDRHEADER=13, ADDRMODE=0,
ADDRTRAILER=40, INITSTRING=274A6F7E93, BLOCK=1;

DEFLIKE EXTYPE=DKU7105, LIKE=DKU7107, DSAMODEL=2041, SUBTYPES=
(KB, KCT, CRT, PRT);
```

```
DEFTERM EXTYPE=DKU7211, DSAMODEL=2045, INTYPE=30,
COMFEAT=CSX25, SUBTYPES=(KB, KCT, CRT, PRT), ADDMODE=VIP,
PROC1=VIP, FEATURES=(AUTOLF, HTAB), PRESCLASS=2, TEXTTYPE=9,
CSETTYPE=2, PSWHIDE=274AF894, PAGECLEAR=0C, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80,
LINEMX=80, MESSLG=6144, FORMSTYPE=7, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, FORWARD=12, TOSTATUS=274AF0A5,
OUTSTATUS=274AF1A5, ADDRHEADER=13, ADDRMODE=0,
ADDRTRAILER=40, INITSTRING=274A6F7E93, BLOCK=1;

DEFLIKE EXTYPE=DKU7205, LIKE=DKU7211, DSAMODEL=204B, SUBTYPES=
(KB, KCT, CRT, PRT);

DEFLIKE EXTYPE=DKU211D, LIKE=DKU7211, DSAMODEL=204E, SUBTYPES=
(KB, KCT, CRT, PRT, DSK) ;

DEFTERM EXTYPE=DTU7171, DSAMODEL=2008, INTYPE=06,
COMFEAT=PAD, SUBTYPES=(KB, KCT, CRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PSWHIDE=0D25, PSWCLEAR=1D1F, PAGECLEAR=0D25, SYSHEAD=0D25,
SYSTRAIL=0D25, DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=128, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;

DEFTERM EXTYPE=DTU7172, DSAMODEL=2009, INTYPE=0D,
COMFEAT=PAD, SUBTYPES=(KB, KCT, CRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PAGECLEAR=2779, SYSHEAD=0D25, SYSTRAIL=0D25, DELCHAR=16,
INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80, LINEMX=80,
MESSLG=128, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=HL6, DSAMODEL=4001, INTYPE=2A, SUBTYPES=(KCT,
CPU), PROC1=VIP, PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PSWHIDE=A1, PAGECLEAR=0D25, SYSHEAD=0D25, SYSTRAIL=0D25,
INBFSZ=256, PAGELG=24, PAGEMX=24, LINELG=80, LINEMX=80,
MESSLG=2048, EMULATE=2033, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=HL61, DSAMODEL=1002, INTYPE=60,
SUBTYPES=(CPU), PROC1=BSC, PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PAGECLEAR=0D25, INBFSZ=256, PAGELG=0, PAGEMX=0,
LINELG=0, LINEMX=0, MESSLG=2048, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;

DEFTERM EXTYPE=HL62, DSAMODEL=1003, INTYPE=62,
SUBTYPES=(CPU), PROC1=BSC, PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PAGECLEAR=0D25, INBFSZ=256, PAGELG=0, PAGEMX=0,
LINELG=0, LINEMX=0, MESSLG=2048, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;
```

```
DEFTERM EXTYPE=HL64, DSAMODEL=1004, INTYPE=61,
SUBTYPES=(CPU), PROC1=ANY, PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PAGECLEAR=0D25, INBFSZ=256, PAGELG=0, PAGEMX=0,
LINELG=0, LINEMX=0, MESSLG=2048, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;

DEFTERM EXTYPE=HL66, DSAMODEL=1005, INTYPE=63,
SUBTYPES=(CPU), PROC1=BSC, PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PAGECLEAR=0D25, SYSHEAD=0D25, SYSTRAIL=0D25,
INBFSZ=256, PAGELG=0, PAGEMX=0, LINELG=0, LINEMX=0,
MESSLG=2048, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=IBM2780, DSAMODEL=2062, INTYPE=60,
SUBTYPES=(CPU), PROC1=BSC, PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, INBFSZ=256, PAGELG=0, PAGEMX=0, LINELG=0,
LINEMX=0, MESSLG=2048, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFLIKE IBM278M, LIKE=IBM2780, DSAMODEL=2063;

DEFLIKE IBM3780, LIKE=IBM2780, DSAMODEL=2065;

DEFTERM EXTYPE=IBM3270, DSAMODEL=2061, INTYPE=50,
SUBTYPES=(KB, KCT, CRT, PRT), PROC1=BSC, PRESCLASS=3,
TEXTTYPE=15, CSETTYPE=0, PSWHIDE=1D4C13, PAGECLEAR=1140403C,
SYSHEAD=114040, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80,
LINEMX=80, MESSLG=2048, FORMSTYPE=3, GKSTYPE=0, FORWARD=40,
ADDRHEADER=11, ADDRTRAILER=13, BLOCK=1;

DEFTERM EXTYPE=IBM3278, DSAMODEL=2064, INTYPE=50,
SUBTYPES=(KB, PRT, KCT, CRT), PROC1=BSC, PRESCLASS=3,
TEXTTYPE=15, CSETTYPE=4, PSWHIDE=1D4C13, PAGECLEAR=1140403C,
SYSHEAD=114040, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80,
LINEMX=80, MESSLG=6144, FORMSTYPE=9, GKSTYPE=0, FORWARD=40,
ADDRHEADER=11, ADDRTRAILER=13, BLOCK=1;

DEFLIKE EXTYPE=IBM3279, LIKE=IBM3278, DSAMODEL=2069;

DEFLIKE EXTYPE=IBM3287, LIKE=IBM3278, DSAMODEL=2067,
SUBTYPES=(KPR, PRT);


DEFTERM EXTYPE=IBM370, DSAMODEL=1008, INTYPE=65,
SUBTYPES=(CPU), PROC1=BSC, PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PAGECLEAR=0D25, INBFSZ=256, PAGELG=0, PAGEMX=0,
LINELG=0, LINEMX=0, MESSLG=2048, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;

DEFTERM EXTYPE=IBM3741, DSAMODEL=2060, INTYPE=64,
SUBTYPES=(CPU), PROC1=BSC, PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PAGECLEAR=0D25, INBFSZ=256, PAGELG=0, PAGEMX=0,
LINELG=0, LINEMX=0, MESSLG=1024, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;
```

```
DEFTERM EXTYPE=KDS7255, DSAMODEL=2036, INTYPE=26,
SUBTYPES=(CRT, DSK), ADDMODE=VIP, PROC1=VIP, PRESCLASS=0,
TEXTTYPE=0, CSETTYPE=0, PSWHIDE=A1, PAGECLEAR=0C,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=1, PAGEMX=1,
LINELG=128, LINEMX=128, MESSLG=1024, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;

DEFTERM EXTYPE=KDS7275, DSAMODEL=2038, INTYPE=2D,
SUBTYPES=(KB, KCT, CRT, PRT, DSK), ADDMODE=VIP, PROC1=VIP,
PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0, PSWHIDE=A1,
PAGECLEAR=0C, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=1, PAGEMX=1, LINELG=128, LINEMX=128, MESSLG=1024,
FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=MINITEL, DSAMODEL=2024, INTYPE=1F,
COMFEAT=PAD, SUBTYPES=(KB, CRT, KCT), PROC1=TTY, PROC2=HDLC,
FEATURES=(WRAP), PRESCLASS=2, TEXTTYPE=11, CSETTYPE=3,
PSWHIDE=1127D7277C4016, PSWCLEAR=27C74027E0, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=40,
LINEMX=40, MESSLG=6144, PAGECLEAR=0C0F, INITSTRING=11,
ADDRMODE=2, ADDRHEADER=1F, ADDRTRAILER=0F, FORMSTYPE=8,
BLOCK=0, GKSTYPE=0, ROLLON=277A89C3, ROLLOFF=277A91C3;

DEFTERM EXTYPE=MINITELB, DSAMODEL=202A, INTYPE=1F,
COMFEAT=PAD, SUBTYPES=(KB, CRT, KCT), PROC1=TTY, PROC2=HDLC,
FEATURES=(WRAP), PRESCLASS=2, TEXTTYPE=16, CSETTYPE=3,
PSWHIDE=1127D7277C4016, PSWCLEAR=27C74027E0, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80,
LINEMX=80, MESSLG=6144, PAGECLEAR=2783, INITSTRING=11,
ADDRMODE=2, ADDRHEADER=274A, ADDRTRAILER=C8, FORMSTYPE=12,
BLOCK=0, GKSTYPE=0, ROLLON=277A89C3, ROLLOFF=277A91C3;

DEFTERM EXTYPE=ARCHITEL, DSAMODEL=2082, INTYPE=1F,
COMFEAT=PAD, SUBTYPES=(KB, CRT, KCT), PROC1=TTY, PROC2=HDLC,
FEATURES=(WRAP), PRESCLASS=2, TEXTTYPE=11, CSETTYPE=3,
PSWHIDE=1127D7277C4016, PSWCLEAR=27C74027E0, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=40,
LINEMX=40, MESSLG=16384, PAGECLEAR=0C0F, INITSTRING=11,
ADDRMODE=2, ADDRHEADER=1F, ADDRTRAILER=0F, FORMSTYPE=8,
BLOCK=0, GKSTYPE=0, ROLLON=277A89C3, ROLLOFF=277A91C3;

DEFTERM EXTYPE=MTS7500, DSAMODEL=2034, INTYPE=24,
SUBTYPES=(KB, KCT, CRT, PRT, CAS, DSK), ADDMODE=VIP,
PROC1=VIP, FEATURES=(AUTOLF), PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PSWHIDE=A1, PAGECLEAR=0D25, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=12, PAGEMX=12, LINELG=80,
LINEMX=80, MESSLG=1024, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;
```

```
DEFTERM EXTYPE=PC7800, DSAMODEL=2029, INTYPE=0E, COMFEAT=PAD,
SUBTYPES=(KB, CRT, KCT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=2, TEXTTYPE=6, CSETTYPE=5,
PSWHIDE=27A2C8, PSWCLEAR=2779, SYSHEAD=0D25, SYSTRAIL=0D25,
DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80,
LINEMX=80, MESSLG=2048, TCTINDEX=0, PAGECLEAR=27D1,
FORMSTYPE=10, GKSTYPE=0, TABCLEAR=274AD5, TABSET=2797,
FORWARD=27C3, TOSTATUS=27A6, OUTSTATUS=27A5,
ADDRHEADER=278640, ADDRMODE=0, ROLLON=2799, ROLLOFF=2798,
INITSTRING=274AC1274A84, BLOCK=1;

DEFTERM EXTYPE=PRT1220, DSAMODEL=205C, INTYPE=07,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=1, TEXTTYPE=13, CSETTYPE=1,
PAGECLEAR=0C, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=66, PAGEMX=00, LINELG=132, LINEMX=132, MESSLG=6144,
FORMSTYPE=0, GKSTYPE=4, TABCLEAR=274AF0A7, TABSET=27D75AF15E,
PLHEADER=27D7E0F17A, PLTRAILER=27E0274AF15EF2A6, PLMODE=3,
INITSTRING=274A6F7E88, BLOCK=0;

DEFTERM EXTYPE=STS2840, DSAMODEL=2051, INTYPE=2F,
SUBTYPES=(KB, KCT, CRT, PRT, DSK), ADDMODE=VIP, PROC1=VIP,
PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0, PSWHIDE=1D76,
PSWCLEAR=0C, PAGECLEAR=0C, SYSHEAD=0D25, SYSTRAIL=0D25,
INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80, LINEMX=80,
MESSLG=2048, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=SWITCH, DSAMODEL=FFFF, INTYPE=FF,
SUBTYPES=(CPU), PROC1=HDLC, FEATURES=(), PRESCLASS=0,
TEXTTYPE=0, CSETTYPE=0, PAGECLEAR=0D25, PAGELG=0, PAGEMX=0,
LINELG=0, LINEMX=0, MESSLG=0, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;

DEFTERM EXTYPE=TCV260, DSAMODEL=2012, INTYPE=40,
SUBTYPES=(KB, KCT, CRT), PROC1=TCV, PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PSWCLEAR=18, PAGECLEAR=18, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=16, PAGEMX=16, LINELG=64,
LINEMX=64, MESSLG=1024, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=TC349, DSAMODEL=2013, INTYPE=41, SUBTYPES=(KB,
KPR, PRT), PROC1=TCV, FEATURES=(ROLL), PRESCLASS=0,
TEXTTYPE=0, CSETTYPE=0, PAGECLEAR=0D25, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=0, PAGEMX=0, LINELG=120,
LINEMX=120, MESSLG=128, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=TC380, DSAMODEL=2014, INTYPE=42, SUBTYPES=(KB,
KPR, PRT), PROC1=TCV, FEATURES=(ROLL), PRESCLASS=0,
TEXTTYPE=0, CSETTYPE=0, PAGECLEAR=0D25, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=0, PAGEMX=0, LINELG=120,
LINEMX=120, MESSLG=128, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;
```

```
DEFTERM EXTYPE=TELEX, DSAMODEL=2021, INTYPE=1E, COMFEAT=PAD,
SUBTYPES=(KB, KPR), PROC1=TTY, PROC2=TTY_R, FEATURES=(ROLL),
PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=0, PAGEMX=0, LINELG=69,
LINEMX=69, MESSLG=128, PAGECLEAR=0D25, FORMSTYPE=0,
GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=TE318, DSAMODEL=2006, INTYPE=05, SUBTYPES=(KB,
KPR, PRT), PROC1=TTY, PROC2=TTY_R, FEATURES=(ROLL),
PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0, PAGECLEAR=0D25,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=0, PAGEMX=0,
LINELG=72, LINEMX=72, MESSLG=128, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;

DEFTERM EXTYPE=TK4105, DSAMODEL=2028, INTYPE=07, COMFEAT=PAD,
SUBTYPES=(KB, CRT, KCT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, AUTOFOLD, HTAB), PRESCLASS=2, TEXTTYPE=12,
CSETTYPE=0, PSWCLEAR=27F2D1, SYSHEAD=0D25, SYSTRAIL=0D25,
DELCHAR=16, INBFSZ=00, PAGELG=30, PAGEMX=30, LINELG=80,
LINEMX=80, MESSLG=2048, PAGECLEAR=274AF2D1, FORMSTYPE=0,
GKSTYPE=3, TABCLEAR=274AF287, TABSET=27C8, FORWARD=274AF1C3,
ADDRHEADER=274A, ADDRTRAILER=5EF1C8, ADDRMODE=2,
PLHEADER=27D3D3, PLMODE=1, INITSTRING=276C4FF027D3E5F027D3C2,
BLOCK=0;

DEFTERM EXTYPE=TN1200, DSAMODEL=2011, INTYPE=0A,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, AUTOLF, HTAB), PRESCLASS=0, TEXTTYPE=2,
CSETTYPE=0, PSWHIDE=275E, PSWCLEAR=277A, PAGECLEAR=2507,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=0, PAGEMX=0,
LINELG=120, LINEMX=120, MESSLG=256, FORMSTYPE=0, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, BLOCK=0;

DEFTERM EXTYPE=TN300, DSAMODEL=2005, INTYPE=04, COMFEAT=PAD,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, AUTOLF, HTAB), PRESCLASS=0, TEXTTYPE=2,
CSETTYPE=0, PSWHIDE=275E, PSWCLEAR=277A, PAGECLEAR=2507,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=0, PAGEMX=0,
LINELG=118, LINEMX=118, MESSLG=128, FORMSTYPE=0, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, BLOCK=0;

DEFTERM EXTYPE=TN340, DSAMODEL=2073, INTYPE=07, COMFEAT=PAD,
SUBTYPES=(PRT, KPR, KB), PROC1=TTY, PROC2=TTY_R,
FEATURES=(AUTOLF, HTAB, VTAB), PRESCLASS=0, TEXTTYPE=2,
CSETTYPE=0, PSWHIDE=275E, PSWCLEAR=277A, PAGECLEAR=2507,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=0, PAGEMX=0,
LINELG=132, LINEMX=132, MESSLG=256, FORMSTYPE=0, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, BLOCK=0;
```

```
DEFTERM EXTYPE=TTS7800, DSAMODEL=203B, INTYPE=29,
SUBTYPES=(KB, KCT, CRT, PRT, DSK), ADDMODE=VIP, PROC1=VIP,
FEATURES=(AUTOLF, HTAB), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PSWHIDE=A1, PAGECLEAR=0C, SYSHEAD=0D25, SYSTRAIL=0D25,
INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80, LINEMX=80,
MESSLG=2048, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=TTU8124, DSAMODEL=200A, INTYPE=07,
COMFEAT=PAD, SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=1, TEXTTYPE=3, CSETTYPE=0,
PAGECLEAR=0C070707, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=66, PAGEMX=0, LINELG=132, LINEMX=132, MESSLG=256,
FORMSTYPE=0, GKSTYPE=0, TABCLEAR=27F2, TABSET=27F1,
PLHEADER=27F0, PLMODE=0, BLOCK=0;

DEFTERM EXTYPE=TTU8126, DSAMODEL=200B, INTYPE=08,
COMFEAT=PAD, SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=1, TEXTTYPE=3, CSETTYPE=0,
PAGECLEAR=0C070707, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=66, PAGEMX=0, LINELG=132, LINEMX=132, MESSLG=256,
FORMSTYPE=0, GKSTYPE=0, TABCLEAR=27F2, TABSET=27F1,
PLHEADER=27F0, PLMODE=0, BLOCK=0;

DEFTERM EXTYPE=TTU8128, DSAMODEL=200C, INTYPE=08,
COMFEAT=PAD, SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PAGECLEAR=0D25, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=66, PAGEMX=00, LINELG=132, LINEMX=132, MESSLG=6144,
FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=TTU8221, DSAMODEL=2039, INTYPE=27,
COMFEAT=CSX25, SUBTYPES=(KB, KPR, PRT), ADDMODE=VIP,
PROC1=VIP, FEATURES=(ROLL, HTAB), PRESCLASS=0, TEXTTYPE=0,
CSETTYPE=0, PAGECLEAR=0C, SYSHEAD=0D25, SYSTRAIL=0D25,
INBFSZ=00, PAGELG=66, PAGEMX=0, LINELG=132, LINEMX=132,
MESSLG=1024, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=TTX35, DSAMODEL=2043, INTYPE=30,
SUBTYPES=(KCT, CRT, PRT, DSK, KB), PROC1=VIP,
FEATURES=(AUTOLF, HTAB), PRESCLASS=2, TEXTTYPE=7, CSETTYPE=0,
PSWHIDE=274AF894, PAGECLEAR=0C, SYSHEAD=0D25, SYSTRAIL=0D25,
DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80,
LINEMX=80, MESSLG=6144, FORMSTYPE=4, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, FORWARD=12, ADDRHEADER=13,
ADDRMODE=0, ADDRTRAILER=40, BLOCK=1;
```

```
DEFTERM EXTYPE=TTY33, DSAMODEL=2001, INTYPE=00, COMFEAT=PAD,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL), PRESCLASS=0, TEXTTYPE=1, CSETTYPE=0,
PAGECLEAR=25, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=0, PAGEMX=0, LINELG=72, LINEMX=72, MESSLG=128,
FORMSTYPE=0, GKSTYPE=0, TABCLEAR=27F2, TABSET=27F1, BLOCK=0;

DEFTERM EXTYPE=TTY35, DSAMODEL=2002, INTYPE=01, COMFEAT=PAD,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL), PRESCLASS=0, TEXTTYPE=1, CSETTYPE=0,
PAGECLEAR=25, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=0, PAGEMX=0, LINELG=72, LINEMX=72, MESSLG=128,
FORMSTYPE=0, GKSTYPE=0, TABCLEAR=27F2, TABSET=27F1, BLOCK=0;

DEFTERM EXTYPE=TTY37, DSAMODEL=2003, INTYPE=02, COMFEAT=PAD,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=0, TEXTTYPE=1, CSETTYPE=0,
PAGECLEAR=25, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=0, PAGEMX=0, LINELG=72, LINEMX=72, MESSLG=128,
FORMSTYPE=0, GKSTYPE=0, TABCLEAR=27F2, TABSET=27F1, BLOCK=0;

DEFTERM EXTYPE=TTY38, DSAMODEL=2004, INTYPE=03, COMFEAT=PAD,
SUBTYPES=(KB, KPR, PRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL, HTAB), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PAGECLEAR=25, SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00,
PAGELG=0, PAGEMX=0, LINELG=72, LINEMX=72, MESSLG=128,
FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=VIP7001, DSAMODEL=203C, INTYPE=2B,
COMFEAT=CSX25, SUBTYPES=(KB, KCT, CRT, PRT), ADDMODE=VIP,
PROC1=VIP, FEATURES=(AUTOLF, HTAB), PRESCLASS=2, TEXTTYPE=4,
CSETTYPE=0, PSWHIDE=A1, PAGECLEAR=0C07, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80,
LINEMX=80, MESSLG=2048, FORMSTYPE=1, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, FORWARD=12, ADDRHEADER=13,
ADDRMODE=0, ADDRTRAILER=40, BLOCK=1;


DEFTERM EXTYPE=VIP7100, DSAMODEL=200D, INTYPE=09,
COMFEAT=PAD, SUBTYPES=(KB, KCT, CRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PSWHIDE=27A2C8, PSWCLEAR=0C, PAGECLEAR=0C, SYSHEAD=0D25,
SYSTRAIL=0D25, DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=1024, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;
```

```
DEFTERM EXTYPE=VIP7200, DSAMODEL=200E, INTYPE=0B,
COMFEAT=PAD, SUBTYPES=(KB, KCT, CRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(ROLL), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PSWHIDE=27A2C8, PSWCLEAR=2779, PAGECLEAR=2779, SYSHEAD=0D25,
SYSTRAIL=0D25, DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=1024, FORMSTYPE=0, GKSTYPE=0,
BLOCK=0;

DEFTERM EXTYPE=VIP7700, DSAMODEL=2033, INTYPE=23,
COMFEAT=CSX25, SUBTYPES=(KB, KCT, CRT, PRT, CAS),
ADDMODE=VIP, PROC1=VIP, FEATURES=(AUTOLF, HTAB), PRESCLASS=2,
TEXTTYPE=5, CSETTYPE=0, PSWHIDE=A1, PAGECLEAR=0C07,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=2048, FORMSTYPE=1, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, FORWARD=12, ADDRHEADER=13,
ADDRMODE=0, ADDRTRAILER=40, BLOCK=1;

DEFTERM EXTYPE=VIP7760, DSAMODEL=203A, INTYPE=28,
COMFEAT=CSX25, SUBTYPES=(KB, KCT, CRT, PRT, DSK),
ADDMODE=VIP, PROC1=VIP, FEATURES=(AUTOLF, HTAB), PRESCLASS=2,
TEXTTYPE=5, CSETTYPE=0, PSWHIDE=A1, PAGECLEAR=0C07,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=2048, FORMSTYPE=2, GKSTYPE=0,
TABCLEAR=27F2, TABSET=27F1, FORWARD=12, ADDRHEADER=13,
ADDRMODE=0, ADDRTRAILER=40, BLOCK=1;

DEFTERM EXTYPE=TXT7801, DSAMODEL=2026, INTYPE=0E,
SUBTYPES=(KB, CRT, KCT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(HTAB), PRESCLASS=2, TEXTTYPE=6, CSETTYPE=5,
PSWHIDE=27A2C8, PSWCLEAR=2779, SYSHEAD=0D25, SYSTRAIL=0D25,
DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24, LINELG=80,
LINEMX=80, MESSLG=2048, TCTINDEX=0, PAGECLEAR=27D1,
FORMSTYPE=5, GKSTYPE=0, TABCLEAR=274AD5, TABSET=2797,
FORWARD=27C3, TOSTATUS=27A6, OUTSTATUS=27A5,
ADDRHEADER=278640, ADDRMODE=0, ROLLON=2799, ROLLOFF=2798,
INITSTRING=274AC1274A85, BLOCK=1;

DEFTERM EXTYPE=VDF7801, DSAMODEL=2027, INTYPE=0E,
SUBTYPES=(KB, CRT, KCT), PROC1=TTY, PROC2=TTY_R, FEATURES=(),
PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0, PSWHIDE=27A2C8,
PSWCLEAR=2779, SYSHEAD=0D25, SYSTRAIL=0D25, DELCHAR=16,
INBFSZ=00, PAGELG=72, PAGEMX=72, LINELG=80, LINEMX=80,
MESSLG=2048, TCTINDEX=0, PAGECLEAR=0D25, FORMSTYPE=0,
GKSTYPE=0, BLOCK=0;
```

```
DEFTERM EXTYPE=VIP7801, DSAMODEL=2017, INTYPE=0E,
SUBTYPES=(KB, KCT, CRT), PROC1=TTY, PROC2=TTY_R,
FEATURES=(HTAB), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PSWHIDE=27A2C8, PSWCLEAR=2779, PAGECLEAR=2779, SYSHEAD=0D25,
SYSTRAIL=0D25, DELCHAR=16, INBFSZ=00, PAGELG=24, PAGEMX=24,
LINELG=80, LINEMX=80, MESSLG=2048, TCTINDEX=0, FORMSTYPE=0,
GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=VIP7804, DSAMODEL=203E, INTYPE=2E,
SUBTYPES=(KB, KCT, CRT, PRT), ADDMODE=VIP7800, PROC1=VIP,
FEATURES=(HTAB), PRESCLASS=2, TEXTTYPE=6, CSETTYPE=5,
PSWHIDE=27A2C8, PSWCLEAR=274AC827D1, PAGECLEAR=27D1,
SYSHEAD=0D25, SYSTRAIL=0D25, INBFSZ=1024, PAGELG=72,
PAGEMX=72, LINELG=80, LINEMX=80, MESSLG=9999, FORMSTYPE=5,
GKSTYPE=0, TABCLEAR=274AD5, TABSET=2797, FORWARD=27C3,
TOSTATUS=27A6, OUTSTATUS=27A5, ADDRHEADER=278640, ADDRMODE=0,
ROLLON=2799, ROLLOFF=2798, INITSTRING=274AC1274A84, BLOCK=1;

DEFTERM EXTYPE=VIP785, DSAMODEL=2032, INTYPE=22,
SUBTYPES=(KB, KCT, CRT), ADDMODE=VIP, PROC1=VIP,
FEATURES=(WRAP), PRESCLASS=0, TEXTTYPE=0, CSETTYPE=0,
PSWCLEAR=0C, PAGECLEAR=0C, SYSHEAD=0D25, SYSTRAIL=0D25,
INBFSZ=00, PAGELG=22, PAGEMX=22, LINELG=92, LINEMX=92,
MESSLG=2048, FORMSTYPE=0, GKSTYPE=0, BLOCK=0;

DEFTERM EXTYPE=HDS7, DSAMODEL=204D, INTYPE=2E, SUBTYPES=(KB,
KCT, CRT, PRT), ADDMODE=VIP7800, PROC1=VIP, FEATURES=(HTAB),
PRESCLASS=2, TEXTTYPE=6, CSETTYPE=5, PSWHIDE=27A2C8,
PSWCLEAR=274AC827D1, PAGECLEAR=27D1, SYSHEAD=0D25,
SYSTRAIL=0D25, INBFSZ=1024, PAGELG=72, PAGEMX=72, LINELG=80,
LINEMX=80, MESSLG=9999, FORMSTYPE=11, GKSTYPE=0,
TABCLEAR=274AD5, TABSET=2797, FORWARD=27C3, TOSTATUS=27A6,
OUTSTATUS=27A5, ADDRHEADER=278640, ADDRMODE=0, ROLLON=2799,
ROLLOFF=2798, INITSTRING=274AC1274A84, BLOCK=1;

DEFLIKE EXTYPE=VIP8800, LIKE=HDS7, DSAMODEL=204F, CSETTYPE=7;

DEFLIKE EXTYPE=TWS2255, LIKE=PC7800, DSAMODEL=2090, SUBTYPES=
(KB, CRT, KCT, PRT), CSETTYPE=7;

DEFLIKE EXTYPE=TXT2255, LIKE=PC7800, DSAMODEL=2090, SUBTYPES=
(KB, CRT, KCT, PRT), CSETTYPE=7;

DEFLIKE EXTYPE=VIP7805, LIKE=VIP7804, DSAMODEL=4002, EMULATE=203E;

DEFLIKE EXTYPE=MT281, LIKE=PRT1220, DSAMODEL=4000, EMULATE=205C;

DEFLIKE EXTYPE=VIP7814, LIKE=VIP7804, DSAMODEL=2047, EMULATE=203E;

DEFLIKE EXTYPE=WEB7, LIKE=DKU7105, DSAMODEL=20A0;

DEFLIKE EXTYPE=TERM-RAW LIKE=TTU8124 DSAMODEL=20A1 CSETTYPE=9;
```

C.9.2 Example of Non-Standard Terminal: MY_TERM_MB

```
DEFLIKE EXTYPE=MY_TERM DSAMODEL=6001 LIKE=TTU8124 PSWHIDE=C1C2C3;
```

UPTERM EXTYPE=TTU8124 PAGECLEAR=6D6D6D;

❑

D. Handling MCS Queues

D.1 Queue Configuration

When MCS is used, information concerning its functions must be supplied at network generation. Queues can only be described by low-level directives.

MCS information involves defining:

- global QMON parameters which are defined at the start of the network generation in the NET directive
- and queue attributes used by MCS, each queue being declared by its QD directive.

D.2 Disk Queue File

When the basic network configuration description contains at least one disk QUEUE declaration, a corresponding queue file must be preallocated and preformatted by NETGEN so that the disk queues can be accessed.

All remote systems which dialog with QMON for MCS applications must be declared in the Basic Network Configuration. Remote systems declared in the Directory Configuration must not include those which dialog with QMON.

D.2.1 Preallocating the Disk Queue File

The disk queue file must be preallocated before running NETGEN as follows:

```
S: BUILD_MCS_FILE 'mydiskqueues:media:devclass'
   EXPDATE = 365
   SIZE    = 5;

S:
```

The file must be monovolume and its size in cylinders must comply with the constraints of its device class and must start with "MS/". When a file is already allocated but its size is not suitable, it must first be deallocated as follows:

```
S: DELETE_FILE 'mydiskqueues:media:devclass';

S:
```

This deallocation is allowed if MAM=NO (default) has been declared at system initialisation. However, the disk file must not be deallocated if:

- either it is already defined for a network currently in use
- or MAM=YES or MAM=REFORMAT has been declared at system initialization, thereby allocating the file to a system process group.

D.2.2 Disk File Preformatting

If the basic network configuration description contains disk queue directives, QFILE=filename must be specified. This file will be automatically preformatted by NETGEN at enabling time (ENABLE=1):

- either just after the generation (CRNET=member)
- or when this same configuration is later loaded (LOAD=member).

Control records are written at the beginning of the file so that the disk queues can be restarted. The control record format depends on the queue attributes defined in the NETGEN directives.

The disk queue file must not be deallocated or reallocated without recreating the network generation.

Memory queues defined in the basic network configuration description are preformatted in main memory.

D.3 NET (Network) Directive

Purpose

This directive applies only to the basic network configuration description. It identifies the network generation and provides global information and specific MCS parameters if needed.

It is optional only if high-level directives are specified. If the network generation contains only low-level directives, the NET directive is mandatory and must be the first specified except COMMENT.

Syntax

```
{ NET     }
{         }        NAME = name4
{ NETWORK }

[  { QM      }   { 0 } ]
[, {         } = {   } ]
[  { GENQMON }   { 1 } ]

[  { SIMU    }   { 1    } ]
[, {         } = {      } ]
[  { MAMSIMU }   { dec2 } ]

[, APPLIB = name-44 ]

[                { 20   } ]
[, RESTARTCNT = {      } ]
[                { dec3 } ]

[  { PSSW     }          ]
[, {          } = name10 ]
[  { PASSWORD }          ]

[             { 76   } ]
[, QMBLKSZ = {      } ]
[             { dec4 } ]

[             { 512  } ]
[, QFBLKSZ = {      } ]
[             { 1024 } ]

[            { 0    } ]
[, QMPOOL = {      } ] ;
[            { dec3 } ]
```

Parameter	Description
NAME	Identifies the current basic network configuration generation. *NAME=name4:* right-padded with underscores (_) if less than 4 characters. No default. If the local system name is omitted in the LSYS directive, the default name is that of the network generation.
QM alias **GENQMON**	Indicates if MCS services are required in the generation. *QM=0:* ***Default*** The current network generation does not concern MCS. The following MCS parameters do not apply. *QM=1:* MCS is required in the current network generation. This means that at least one QD directive must be specified. Parameters dealing exclusively with MCS are validated.
APPLIB	**Related to MCS**. Defines the name of the cataloged application library whose members are referenced in INITJCL of the QD directive. *APPLIB=lib44*: No default.
RESTARTCNT	**Related to MCS**. Defines the number of retries in starting up an MCS application, on abnormal termination, to prevent the system from indefinitely spawning an incorrect application. 0 <= *restartcnt* <= 100. ***Default:*** is 20.
SIMU alias **MAMSIMU**	**Related to MCS**. Defines the maximum number of MCS processes that can execute simultaneously. 1 <= *simu* <= 64. ***Default:*** 1.
PSSW alias **PASSWORD**	**Related to MCS**. *PSSW=name10:* Defines the unique password used by MCS applications when executing ENABLE and DISABLE verbs, as specified with the KEY parameter of these verbs. No checking will be performed when these verbs are executed if this parameter was not supplied in the NET directive.

QMBLKSZ

Related to MCS. Defines the size in bytes of each block in the memory queue pool. *This parameter should not be specified if no memory QUEUEs are declared.*

76 <= *qmblksz* <= 1024. Actual usable size = (*qmblksz* - 5).
Default*: 76.*

QFBLKSZ

Related to MCS. Defines the size in bytes of the buffer unit determining the block size of the disk queue file. *This parameter must not be specified if disk QUEUEs are not declared.*
*QFBLKSZ=512 (****default****)* or *QFBLKSZ=1024*.
Actually usable size is (*qfblksz* - 5).

QMPOOL

Related to MCS. Defines the number of memory queue pool blocks to be shared by all memory queues with QMPOOL=1 in the associated QD directive.

0 <= *qmpool* <= 862.
QMPOOL=0: ***Default***
no QD directive may specify QMPOOL=1.
The maximum number of memory blocks is defined by:
*((qmpool + total (memblock)) * qmblksz) <= 65535, where*
MEMBLOCK values are specified in the QD directives.

D.4 QD (Queue) Directive

Purpose

This directive applies only to the basic network configuration description. It defines MAM queue attributes and applies only if the NET directive specifies option QM. A maximum of 255 queues can be declared.

Syntax

```
{ QD    }                                [       { PROGRAM | PG   } ]
{       }        NAME=char12             [,TYPE={ DSA            } ]
{ QUEUE }                                [       { USER | USERID } ]

[  { ST    }   { ENBL } ]                { [  { MBLK     }        ]  }
[, {       } = {      } ]                { [, {          } = dec3 ]  }
[  { STATE }   { LOCK } ]                { [  { MEMBLOCK }        ]  }
                                         {                           }
[  { INIT    }          ]                { [  { FBLK      }        ] }
[, {         } = name12 ]                { [, {           } = dec5 ] }
[  { INITJCL }          ]                { [  { FILEBLOCK }        ] }
                                         {                           }
[  { ENQ     }   { 0 } ]                 { [           { 0 } ]       }
[, {         } = {   } ]                 { [, QMPOOL = {   } ]       }
[  { ENQUEUE }   { 1 } ]                 { [           { 1 } ]       }

[ { BRK   } { 0 } ]     [       { 0 } ]     [ {USER   } { 255  } ]
[,{       }={   } ]     [,SHARE={   } ]     [,{       }={      } ]
[ { BREAK } { 1 } ]     [       { 1 } ]     [ {USERNB} { dec3 } ]

[ { OM      } { NORMAL   | NL }]         [ { IM     } { NORMAL   | NL }]
[,{         }={               }]         [,{        }={ MARK     | MK }]
[ { OUTMODE} { UNEDITED | UN }]          [ { INMODE } { UNEDITED | UN }]

[     { 0 } ]                            [ { BLKG    } { 0 } ]
[,TWA={   } ]                            [,{         }={   } ]
[     { 1 } ]                            [ { BLOCKING} { 1 } ]

[         { NO         } ]               [ { LNLG     } { 80   } ]
[,RESTART={ CONTROLLED } ]               [,{          }={      } ]
[         { ROLLBACK   } ]               [ { LINELGTH } { dec4 } ]

[ { NBLN    } { 20   } ]                 [        { 0 } ]
[,{         }={      } ]                 [,BSCACK={   } ] ;
[ { NBLINES } { dec4 } ]                 [        { 1 } ]
```

Parameter	Description
NAME	Length and syntax depend on the queue type: *NAME=name8:* **Program-queue** input queue used by the MCS application to receive messages. *NAME=name12:* **Userid-queue** output queue for a terminal used when the userid given at logon in a switched network corresponds to the queue name. *NAME=name4.name8*: **DSA-queue** queue used to send messages to a terminal or to an MCS application on a remote DPS 7000, of the format: *<session_control_name>.<mailbox_name>*, where *session_control_name is* defined through the RSC directive for either terminals configured to the Datanet or CNP 7, or links to the remote system. $ can optionally prefix the name. *mailbox_name* identifies either the terminal in DNS or CNS 7 SYSGEN, or the remote application being the local queue in the remote DPS 7000. $ can optionally prefix the name.
TYPE	Corresponds to the 3 preceding cases of NAME. *TYPE=PG or PROGRAM:* ***Default*** *TYPE=USER or USERID* *TYPE=DSA*
ST alias **STATE**	Defines the initial state of the queue: *ST=ENBL:* ***Default*** Data transfers occur immediately at startup. *ST=LOCK:* The ENABLE verb must first be issued and executed before data can be transferred.

MBLK alias **MEMBLOCK**

MEMBLOCK, FILEBLOCK and QMPOOL=1 are mutually exclusive. Defines a memory queue and specifies the number of memory blocks for the exclusive use of the queue.
Size of a memory block and the number of memory blocks shared by all MCS applications are defined by the QMBLKSZ and QMPOOL parameters of the NET directive.
The maximum memory size is:
*((total (memblock) + qmpool) * qmblksz)*
<= 65535 bytes.
2 <= memblock <= 862. No default.

FBLK alias **FILEBLOCK**

MEMBLOCK, FILEBLOCK and QMPOOL=1 are mutually exclusive. Defines the queue as a disk file queue and specifies the number of disk queue file blocks for the exclusive use of the queue. Block size is defined in the QFBLKSZ parameter of the NET directive.
2 <= *fileblock* <= 32767. No default.

QMPOOL Defines if the queue can access the common memory pool:
QMPOOL=0: ***Default***
The queue can use private memory defined by MEMBLOCK in the same QD directive to protect a memory pool from being exhausted by excessive output from an application.

QMPOOL=1: **MEMBLOCK, FILEBLOCK and QMPOOL=1 are mutually exclusive.** The queue is a memory queue which can only use the common memory pool defined previously by the QMPOOL=dec3 parameter in the NET directive.

RESTART Applies only to file queues after *step abort* or *system crash*.

RESTART=NO: ***Default***, No file queue recovery.

RESTART=ROLLBACK: **Mutually exclusive with INITJCL.** The file queue is to be rolled back.

RESTART=CONTROLLED: **Mutually exclusive with INITJCL and not applicable to program-queues**. A controlled restart is possible.

INIT alias **INITJCL**	**Mutually exclusive with RESTART=ROLLBACK and RESTART=CONTROLLED**. *INIT=name12:* Defines the JCL subfile in the cataloged application library which must have been specified in APPLIB of the NET directive. See *MCS User's Guide*.
BRK alias **BREAK**	Defines if the application associated with the queue is to be notified of: asynchronous events such as *BREAK* and *RVI* and terminal status changes such as connection and disconnection. *BRK=0:* ***Default***, the application is not notified of these events. *BRK=1*
ENQ alias **ENQUEUE**	Specifies the action taken by the GCOS 7 session layer when the maximum number of connections (USERNB) to the program is reached. *ENQ=0:* ***Default***, Rejects new connection requests. *ENQ=1:* Enqueues all further connection requests until a disconnection occurs.
USER alias **USERNB**	**Applicable only to program-queues.** Defines the maximum number of users that can simultaneously connect to the program queue. 1 <= *usernb* <= 255. ***Default:*** 255. ENQUEUE defines the action to be taken when the number of users (connection requests) exceeds this value.
SHARE	**Applicable only to program-queues**. Specifies if the queue can be shared. *SHARE=0:* ***Default***, The queue cannot be shared. *SHARE=1:* Two or more MCS applications can share the same queue with either input or input/output capability defined in their QASSIGN statement. *Should not be used if the queue is assigned as a subqueue in the JCL QASSIGN statement.*

TWA	**Applicable only to program-queues**. *Two Way Alternate* protocol, where: the application can issue 1 or more SENDs WITH EMI and keeps the turn the application relinquishes the turn on SEND WITH EGI at the end of a message sent by the terminal, the turn is given over to the application at logon, the terminal has the initial turn. *TWA=0:* ***Default***, no TWA protocol, therefore *Two Way Simultaneous* protocol is used. *TWA=1:* TWA protocol is used.
BLKG alias **BLOCKING**	**Not applicable to program-queues.** Defines if editing is to be performed by MAM. *BLKG=0:* ***Default***, No such editing by MAM. *BLKG=1:* MAM edits messages sent by the application to the terminal by prefixing an NL (new-line) and CR (carriage-return) control code to each line of the message text.
LNLG alias **LINELGTH**	**Not applicable to program-queues and applies only if BLOCKING is specified**. Defines the logical line length for the terminal, for automatic editing. 5 <= *linelgth* <= 9999. ***Default:*** 80.
NBLN alias **NBLINES**	**Not applicable to program-queues and applies only if BLOCKING is specified.** Defines the maximum number of logical lines which can be sent in a message to the terminal. The maximum length of the message is: *(nblines * linelgth)*. 1 <= *nblines* <= 9999. ***Default:*** 20.

IM alias **INMODE**

Not applicable to program-queues. Specifies the format of messages passed from the terminal to the MCS application:

IM=NORMAL: ***Default***

headers and control codes, except HT and ESC, are not passed
character encoding and repeat formats are translated
other characters, including control codes in mark format, are passed without translation.

IM=MARK:

headers are passed as marks of the format ><U03abc
all hexadecimal control codes are translated into their corresponding mark formats
character encoding and repeat formats are translated
other characters, including control codes in mark format, are passed without translation.

IM=UNEDITED:

headers are passed as marks of the format ><U03abc
other characters, including control codes in mark format, character encoding and repeat formats are passed without translation.

OM alias **OUTMODE**

Not applicable to program-queues. Specifies the format of data passed from the MCS application to the terminal:

OM=NORMAL: ***Default***
- the user can provide VIP-headers in the mark format ><U03abc
- character encoding and repeat formats are translated
- unreserved codes and unapplicable hexadecimal control codes are translated into character encoding format ><Cab
- control codes in mark format specific to the terminal are translated into hexadecimal values
- all other characters, including applicable hexadecimal control codes, are passed without translation.

OM=UNEDITED:
- the user can provide VIP-headers in the mark format ><U03abc
- other characters, including control codes in mark format, character encoding and repeat formats are passed without translation.

BSCACK

Applicable only using BSC2780 protocol. Defines if each message exchanged between the DPS 7000 and the Datanet is to be acknowledged.

BSCACK=0: ***Default***, No acknowledgement.
BSCACK=1: Each message exchanged is ACKed

D.5 Examples of Queue Descriptions

I- SCHEMATIC DESCRIPTION

The following example does not reflect an actual MCS application, but is intended only to show how several types of queues can be described and inserted in the basic network configuration described in either in low-level or high-level directives:

II- LOW-LEVEL DIRECTIVES

```
NET PP50 QM QFBLKSZ = 1024, QMPOOL = 16;

COMM ' ------------------------------------------- ';
COMM '|   L O C A L   S Y S T E M : D P S 7 / 7000 |';
COMM ' ------------------------------------------- ';
LSYS BP50     PF='DPS7/GCOS-7/V10';
LSC  BP50     SCID=60:57;
SVR  FE3F     TYPE=FEPS LCT=CC01;
LCT  CC01     TYPE=PSI WATCH=2000;
SVR  TNS;
LCT  EA11;
LPL  PLINK_01 LCT=EA11 IADDR=08-00-38-50-3C-39
              CB=CBL_01;
CP   CP_01    LPL=PLINK_01;

COMM ' ------------------------------------- ';
COMM '| PASSTHROUGHS, NEIGHBORS AND REMOTE   |';
COMM '| SYSTEMS ARE DESCRIBED IN EXAMPLE 1   |';
COMM ' ------------------------------------- ';
```

```
COMM ' ------------------------------------- ';
COMM '|      M E M O R Y    Q U E U E S      |';
COMM ' ------------------------------------- ';
QUEUE QMM_PG01                    MEMBLOCK = 10;
QUEUE QMP_PG02     TYPE = PG      QMPOOL
                                  SHARE = 1, TWA;
QUEUE QMP_US03     TYPE = USER QMPOOL, OM = UNEDITED;
QUEUE BP50.mailbox TYPE = DSA  MEMBLOCK = 50
                                  STATE = LOCK, BLOCKING;

COMM ' ------------------------------------ ';
COMM '|       D I S K      Q U E U E S      |';
COMM ' ------------------------------------ ';
QUEUE QD_PG01;
QUEUE QD_PG02 TYPE = PG   FILEBLOCK = 100
                          USER=10, ST = LOCK;
QUEUE QD_US03 TYPE = USER FILEBLOCK = 50
                          LINELGTH = 160, NBLINES = 12;
QUEUE BP50.mbx TYPE = DSA FILEBLOCK = 25
                          BLOCKING, IM = MARK;

ENET;
```

III- HIGH-LEVEL DIRECTIVES

```
NET BP50 QM QFBLKSZ = 1024, QMPOOL = 16;

COMM ' -------------------------------------------- ';
COMM '|  L O C A L  S Y S T E M  :  D P S 7 / 7000  |';
COMM ' -------------------------------------------- ';
SYS  PP50 PF=LSYS SCID=60:57 ISL=(50-3C-39,EA11);

COMM ' ------------------------------------- ';
COMM '| PASSTHROUGHS, NEIGHBORS AND REMOTE   |';
COMM '| SYSTEMS ARE DESCRIBED IN EXAMPLE 1   |';
COMM ' ------------------------------------- ';
```

```
COMM ' ------------------------------------- ';
COMM '|      M E M O R Y    Q U E U E S      |';
COMM ' ------------------------------------- ';
QUEUE QMM_PG01                 MEMBLOCK = 10;
QUEUE QMP_PG02     TYPE = PG   QMPOOL
                               SHARE = 1, TWA;
QUEUE QMP_US03     TYPE = USER QMPOOL, OM = UNEDITED;
QUEUE BP50.mailbox TYPE = DSA  MEMBLOCK = 50
                               STATE = LOCK, BLOCKING;

COMM ' ------------------------------------ ';
COMM '|       D I S K      Q U E U E S      |';
COMM ' ------------------------------------ ';
QUEUE QD_PG01;
QUEUE QD_PG02 TYPE = PG   FILEBLOCK = 100
                          USER=10, ST = LOCK;
QUEUE QD_US03 TYPE = USER FILEBLOCK = 50
                          LINELGTH = 160, NBLINES = 12;
QUEUE BP50.mbx TYPE = DSA FILEBLOCK = 25
                          BLOCKING, IM = MARK;

ENET;
```

E. Migration from DSA to ISO

DSA source configuration descriptions prepared for the GCOS 7 release V6 remain fully compatible even when ISO support is installed on this GCOS 7 V9 release. However, despite being compatible, all the network configuration descriptions must be submitted for recompilation by the NETGEN utility because the previous binary configurations do no longer apply once ISO support is installed.

To ensure compatibility, DSA addressing and protocol arguments are always supplied by default.

Migration to ISO always involves explicitly adding the corresponding ISO arguments in the source configuration descriptions before recompilation by NETGEN.

NOTES:

After ISO support has been installed on the V6 release,

1. Some RSC directives may conflict by TYPE with the SYSTEM directive which they are to override. To correct this anomaly, TYPE specified in the SYSTEM directive must be *added* to the conflicting RSC directive. See Expansion Overriding in Section 7.
2. Any attempt to *enable* a configuration generated under V6 prior to ISO installation, will be rejected with the following severity 4 error message:

   ```
   ***NG030 UNABLE TO LOAD FROM BINLIB:
               OBSOLETE {NETWORK | DIRECTORY | TERMINAL}
   ```

   The same error message will appear if a configuration generated on a V6 system on which ISO support has been installed, is enabled on another V6 system prior to ISO installation.

❑

Glossary

A

ADDRESSING PID

It is the part of the PID module which is in charge of converting the session addressing from DSA to ISO, and vice versa.

ADMINISTRATIVE CORRESPONDENT

It is the local representation of a remote network administrative function whose network location is specified by associated session control and mailbox names. When statically configured, it is located in the basic network configuration.

B

BASIC NETWORK CONFIGURATION

Part of the network configuration which must always be present and cannot be swapped during a communications session.

BASIC NETWORK CONFIGURATION DESCRIPTION

Set of directives describing the local system, its passthroughs and neighbors and possibly some remote systems. It may also comprise network administration and queue descriptions. It is embedded between NETWORK and ENDNETWORK directives.

BASIC NETWORK DICTIONARY

Dictionary which contains the names and locations of all communications objects statically configured and produced by a basic generation.

BASIC NETWORK GENERATION

NETGEN processing applied to a basic network configuration description. This processing can normally be executed while a communications session is running. However, its enabling requires the communications session to be previously stopped.

C

COMMUNICATIONS SESSION

The activity state of the configurations enabled in the running workspace.
The session is considered *in progress* if at least:

- one communications server is active in the *used* state
- one network operator command is still executing
- one application such as TDS or MCS is still using VCAM.

Otherwise the session is considered *completed*.

CONFIGURATION

Data resulting of the NETGEN generation processing applied to configuration description directives. A full set of configurations comprises:

- the basic network configuration
- the directory configuration
- and the terminal configuration.

CORRESPONDENT

In general terms, it is a local representation of a remote object, function or application associated with a name and a type, and network references to access it.

CORRESPONDENTS DICTIONARY

Dictionary which contains all the objects dealing with the communications service of correspondents which are produced by a directory generation.It can be empty.

D

DICTIONARY

Container filled with information related to each communications object generated. There are 3 different dictionaries:

- the basic network dictionary (DDICT)
- the correspondent dictionary (TCORR)
- the remote system dictionary (TSYS),

the last two forming the directory.

DIRECTIVE

A statement describing one communications object or system and its relationship with others. When describing an object rather than a system it is then called low-level directive.

DIRECTORY

It consists of two dictionaries:

- the remote-system dictionary (TSYS)
- the correspondent dictionary (TCORR).

DIRECTORY CONFIGURATION

Part of the network configuration which is optional and moreover can be fully changed or suppressed during a communications session. It always contains the two dictionaries even if they are empty.
A basic network generation without directory configuration description results in an empty directory configuration available for any subsequent incremental generation.

DIRECTORY CONFIGURATION DESCRIPTION

It must be embeded between DIRECTORY and ENDDIRECTORY directives. It is mainly composed of two parts in any order.

Correspondent Description:

Contains all the objects dealing with the communications service of correspondents. This description is optional, when present the generated objects are always all included into the correspondent dictionary (maximum 10000 correspondents).

Remote-system Description:

Contains all or some remote systems up to 5000, accessible by the local system using passthroughs already described in the basic configuration.

DIRECTORY GENERATION

NETGEN processing applied to a DIRECTORY configuration description. It can be:

- either associated with a basic generation, in which case it is enabled when the basic generation is enabled
- or launched separately, in which case it is *incremental* and its enabling requires a valid basic configuration to be already enabled.

Directory generation may result in an empty directory if the directory configuration descrition is empty.

DYNAMIC CONFIGURATION

Communications objects related to a remote system described within the directory configuration are simply registered in the remote system dictionary but they are dynamically configured and used at connection request time. They remain available until the end of the connection.

E

EMPTY DIRECTORY

This term identifies the result of an empty directory generation, where both the remote system and the correspondent dictionaries are empty. Such directory always allows further *incremental* generations.

EMPTY DIRECTORY CONFIGURATION

Directory configuration which consists in an empty directory.

EMPTY DIRECTORY CONFIGURATION DESCRIPTION

It comprises a directory configuration description without directory objects described between the DIRECTORY and ENDDIRECTORY directives.

EMPTY DIRECTORY GENERATION

NETGEN processing resulting in an empty directory which is applied in the following circumstances:

- to an empty directory configuration description which may be useful when performing *incremental* disabling of a directory already enabled
- in association with a basic network generation specifying CRDIR=NO in the CRNETGEN command

after an abnormal termination of an *incremental* enabling.

ENABLING

GCOS 7 restart time or NETGEN processing which consist in moving the generated or loaded configurations from the permanent workspace into the running workspace.

At GCOS 7 restart CLEAN the enabling is done upon operator's request and it is then preceded by an attempt of loading from the system library members.

F

FRONT-END PROCESSOR

See pass-through

H

HETEROGENEOUS SYSTEM

System which can exchange administrative information in the form of commands and responses, and messages with the local system only in AEP format. Such a system can be any including another DPS 7000 running on releases prior to GCOS 7-V6.

HOMOGENEOUS SYSTEM

System which can exchange administrative information in the form of commands and responses, and messages with the local system in GCOS format with the exception of AUTs which always use AEP format. Such a system can only be a DPS 7000 running on releases from GCOS 7-V6 onwards.

I

INCREMENTAL CONFIGURATION
This processing is specific to OCS. It allows you to dynamically add new remote systems and routes to a running OCS Front End configuration (that is, without having to stop and re-start the OCS server).

INCREMENTAL DISABLING
NETGEN processing applied to an empty directory configuration and launched while a valid basic network configuration is already enabled whether or not the communications session is in progress. This action consists of substituting the generated empty directory to any currently enabled one. *Incremental* enabling which aborts but recovers normally, results in *incremental* disabling since the enabled directory is empty.

INCREMENTAL ENABLING
NETGEN processing applied on a directory configuration description alone and started while a valid basic network configuration is already enabled irrespective of whether a communications session is in progress. The currently generated non-empty directory is replaced with one already enabled one even if it is empty. Such enabling cannot be associated with loading and therefore cannot occur at GCOS 7 restart.
When *incremental* enabling aborts, *incremental* disabling is performed.

INCREMENTAL GENERATION
NETGEN processing applied on a DIRECTORY configuration description alone and executed while a basic configuration is already enabled.
Such generation can be associated with an *incremental* saving and/or an *incremental* enabling.

INCREMENTAL SAVING
NETGEN processing associated with an *incremental* generation and which involves saving into the specified binary library members from the permanent workspace:

- the directory configuration which has just undergone *incremental* generation
- and the basic network and terminal configurations executing in the current communications session.

ISO-DSA PLUG
See PID.

L

LOADING
NETGEN processing which involves moving a full set of configurations from specified binary library members into a permanent or temporary workspace.
Automatic GCOS 7 restart CLEAN moves a full set of configurations from the system binary library members into the permanent workspace.

LOCAL SYSTEM
The unique DPS 7000 system running on GCOS 7 versions from V6 onwards for which the local view of the network is generated. It must be described before any other system.

LOW-LEVEL DESCRIPTION
The network configuration completely or partially described using exclusively low-level directives.

LOW-LEVEL DIRECTIVE
The level of description in which one directive pertains to one communications object.

N

NEIGHBOR
Any system ***directly*** accessible by the local system and linked to the local system either by a PSI channel or over a ISL cable.

NETWORK
The most general term describing intercommunication between nodes and nodes, and nodes and end-users.
The term may also have the more restrictive scope of the ISO network layer environment.

NETWORK ADMINISTRATION
Set of communications functions which regulate, monitor and control events in the network.

NETWORK ADMINISTRATION DESCRIPTION
The part of the basic network description which declares local and remote administrative objects. This description is optional since default administrative objects are implicitly supplied.

NETWORK CONFIGURATION
The resultant data output by the NETGEN utility from the input network description directives. It is composed of:

- the basic network configuration
- and the directory configuration.

NETWORK CONFIGURATION DESCRIPTION
Set of directives fully describing all the communications objects involved within the whole network which are accessible by the local system.
It always comprises a basic network configuration description and where applicable, a directory configuration description.
The basic network configuration description describes the view of the network from the standpoint of the local DPS 7000 which accesses all the systems in the network either directly or by means of other systems. This description is mandatory and can contain up to 512 systems.
All or some of the remote systems, up to 5000, may be described separately in the remote system dictionary which forms part of the directory configuration.

NODE
Synonymous with system.

O

OCS

OCS (Open Communications Subsystem) is the communications driver associated with the OCS Front Ends. It enables a DPS 7000 to access to an FDDI or a high range Ethernet network. OCS Front End contains both an OSI stack and an IPS stack. The OSI stack contains OSI layers 1 to 4 and serves both OSI and DIWS. The IPS stack contains the TCP, UDP, ICMP and IP layers. An RFC1006 layer allows OSI/DIWS session traffic to be carried over a TCP/IP stack.

P

PASSTHROUGH

A neighbor acting as the intermediary through which the local system accesses remote systems or other networks, and vice versa. Such a system frontends the local system and is the *go-between* for remote systems. From GCOS 7-V6, the only passthroughs available for the DPS 7000 are the Datanet and the CNP 7.

The difference between the Datanet and the CNP 7 is that the CNP 7 can only be attached to the same ISL cable as the DPS 7000, whereas the Datanet can be attached either to the ISL cable or through a PSI channel.

PERMANENT WORKSPACE

Set of tables located in the backing store which contains the initial values of all the configurations presently enabled or eligible for enabling, in the running workspace. Since the permanent workspace is used for reinitializing the running workspace at enabling time, it must ***not*** be altered during a communications session by any server.

The contents of this workspace is only destroyed by a GCOS 7 restart CLEAN or by the enabling processing of NETGEN.

PID

The ISO/DSA plug module which interfaces the VCAM session and the TNS/FEPS transport servers to allow DSA applications to communicate in a transparent way with ISO or DSA services.

PROTOCOLAR PID

The part of the PID module in charge of converting session protocols from DSA to ISO, and vice versa.

Q

QUEUE DESCRIPTION

The part of the network generation which describes queues used by the MCS communications processor. The queue description is optional and when present, forms part of the basic network configuration.

R

REMOTE SYSTEM
Any system which the local system accesses through the intermediary of a passthrough. The directive which describes it must appear after all the passthroughs.

REMOTE SYSTEM DICTIONARY
The container for the remote systems which can be installed through *incremental* generation. It can be empty.

RFC1006
This is the reference document which describes the communications layer which enables OSI/DIWS session traffic to be carried on top of TCP/IP protocol. By extension, RFC1006 is also the name of the layer which provides the functionality.

RUNNING WORKSPACE
The set of tables shared at system level and directly accessible by all the communications servers. Valid configurations are stored in it by the enabling function either by the NETGEN utility or at GCOS 7 restart time.

S

SAVING
NETGEN processing which involves moving a full set of configurations from a permanent or temporary workspace into specified binary library members.

STATIC CONFIGURATION
Communications objects described within the basic network configuration are said to be statically configured because they are immediately available to the communications servers at connection request time.
Enabling of such configuration requires the termination of the communications session.

SYSTEM
The functional position occupied by the system within the current network configuration determines its functional type, namely, local system, passthrough, neighbor or remote system.

T

TEMPORARY WORKSPACE
Set of tables private to the NETGEN processor which are used to handle the configurations when enabling is not requested. Using this workspace does not alter the information related to the communications session *in progress*

TERMINAL CONFIGURATION
The terminal configuration description output by the NETGEN utility from the standard terminal description and optional user-defined additions

TERMINAL CONFIGURATION DESCRIPTION

The description of the terminals available to the DPS 7000 site composed of two parts, each in a separate library member but using the same set of directives:

- the standard description which is mandatory, site-specific and unalterable, and located in the system library member: SYS.HSLLIB..H_TERM
- and a user-defined description which adds additional descriptions to standard ones.

TERMINAL GENERATION

NETGEN processing applied to the terminal configuration description. It is always implicitly associated with the basic network generation so that it is automatically started if the terminal configuration expected is absent from the targeted workspace.

TM CORRESPONDENT

The end-user logged on at a terminal to an application such as TDS or IOF, and configured in a directory, *correspondent dictionary*. The correspondent is identified by a name and is defined by a specific network address being the names of its session control and its mailbox.

TP_NODE

The main entity which handles, for a given cooperating application, all the resources necessary for inter-program communication using the XCP2 protocol.

X

XCP1 CORRESPONDENT

A remote application which uses XCP1 protocol in cooperating communication. It is defined by a specific name, is identified by a specific network address being the names of its session control and its mailbox, and is determined by its profile. From GCOS 7-V6, this correspondent is configured in the directory, *correspondent dictionary*.

XCP2 CORRESPONDENT

One of two entities dialoging by means of XCP2 protocol, the one being attached to the local XCP2 workstation, the other being attached to the partner XCP2 workstation. It is defined by a specific name, identified by a specific network address being the names of its session control and its mailbox, and is determined by its profile. This correspondent is configured in the directory, *correspondent dictionary*.

XCP2 WORKSTATION

Functional equivalent of TP-node when used in the NETGEN environment.

List of abbreviations

Acronym	**Meaning**
AC	Administrative Correspondent (ADMCOR) (DSAC)
AEP	Administrative Exchange Protocol (DSAC)
AET	Application Entity Title (OSI addressing)
AF	Administrative Function (ADMFUNC) (DSAC)
AFI	Authority and Format Identifier (OSI addressing)
AG	Administrative Group (ADMGROUP) (DSAC)
API	Application Programatic Interface
APL	A Programming Language (terminal character set)
APPC	Advanced Program to Program Communication
ARP	Address Resolution Protocol (network layer)
ASCII	American Standard Code for Information Interchange
ASF	Administrative Storage Facility (DSAC)
ATS	Applicative Transport Service (ISO/3X7 transport)
AUPI	Administrative Utilities Programmatic Interface (DSAC)
AUT	Administrative UTility (DSAC)
BSC	Binary Synchronous Communication (line procedure)
BSR	Basic System Release
BTNS	Basic Terminal Network Support (V2 comms server)
CAM	Communications Access Method
CCE	Channel Command Enter
CCnn	Communications Controller for PSI of Datanet
CCITT	International Consultative Committee for Telegraph and Telephone
CLnn	Communications Link (SRST entry for CCLINK PSI)
CMD	Command, DSAC concept

Acronym	Meaning
CNC	Communication Network Configurator (V2)
CNP7	Communication Network Processor for the DPS 7000
CNS7	Communication Network Software for CNP7
CP	Communications Path (COMPATH)
CPDS	Communication Processor Dump Scanner
CPU	Central Processor Unit
CRNG	CReate_Network_Generation GCL command (CRNETGEN)
CRT	Cathodic Ray Tube (terminal display feature)
CSA	Communications Server Administration
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
CSPS	Character Set PSeudo-graphics (terminal attribute)
DCC	1.Data Country Code (ISO NSAP addressing)
	2.Data Communication Controller
DDICT	basic network Data DICTionary
DIWS	DSA-ISO Work Station
DJP	Distributed Job Processing
DMAT	Direct Memory Access Transport
DN	DataNet
DNS	Distributed Network Supervisor (Datanet software)
DPS	Distributed Processing System (GCOS)
DPX	Distributed Processing System (UNIX)
DSA	Distributed Systems Architecture
DSAC	Distributed Systems Administration and Control
DSP	Domain Specific Part of NSAP (OSI addressing)
DTE	Data Terminal Equipment

Acronym	Meaning
EAnn	Ethernet Adaptor (TNS or OCS)
EBCDIC	Extended Binary Coded Decimal Interchange Code
EFEP	Extended Front End Processor
FCS	Frame Check Sequence
FDDI	Fiber Distributed Data Interface
FECM	Front End Control Manager (communications server)
FEP	Front End Processor
FEPS	Front End Processor Support (communications server)
FL	Administrative Filter (ADMFILTER) (DSAC)
FNPS	Front-end Network Processor Support (V2 comms server)
FPG7	common FEP and DPS7 interactive generation utility
FTAM	File Transfer Acces Method
GCOS	General Comprehensive Operating System
GKS	Graphics Kernel System
HCB	Header Control Block (MAM)
HDLC	High-level Data Link Control
IC	Integration Component (within SIU)
ICMP	Internet Control Message Protocol
IDI	Initial Domain Identifier of NSAP (OSI addressing)
IDP	1.Initial Domain Part of NSAP (OSI addressing)
	2.ISO / DSA Plug (DSA to ISO migration)
IEEE	Institute of Electrical and Electronic Engineers
IOF	Interactive Operator Facility
IP	Internet Protocol (network layer)

Acronym	Meaning
IPS	Internet Protocol Suite (UDP-TCP/IP)
ISDN	Integrated Service Digital Network
ISL	Inter System Link (applicable to DPS7000)
ISO	International Standards Organization
JOR	Job Occurrence Report
LAN	Local Area Network
LCT	Local Controller (LOCCTLR) named EAnn or CCnn
LFA	Log File Analyzer
LG	Administrative Log (ADMLOG) (DSAC)
LLC	Logical Link Control (link sub-layer for LAN)
LNA	Local Network Adaptor
LNM	Local Network Manager (for DPS 7000 Ax)
LNI	Local Network Interface (for DPS 7000 4xx)
LPL	Local Physical Link (LOCPLINK)
LSAP	Link Service Access Point (OSI addressing)
LSC	Local Session Control (LOCSESS)
LSYS	Local System (LOCSYSTEM)
LT	Logical Terminator (LTERMINATOR)
LTS	Local Transport Station, addressed by TSAP
LU	(6.2)Logical Unit (version 6.2)
	(equivalent to XCP2 within SNA environment)
MAC	Media Access Control (link sub-layer for LAN)
MAM	Message Access Method
MCS	Message Control System
MPC	Multiplexed Peripheral Controller (for DPS7 1x07|10x7|5x0|7x0)

Acronym	Meaning
MUX	administrative MUltipleXor (DSAC)
NAD	Network ADministration subsystem (DSAC)
NCC	Network Control Center (DSAC)
NER	Network Event Reporting (DSAC)
NG	Network Generation utility for DPS7 (NETGEN)
NIP	Non Impact Printer (e.g.: Mathilde)
NMF	Network Management Facility
NNP	New Network Processor
NOI	Network Operator Interface (DSAC)
NPDU	Network Protocol Data Unit (ISO/DSA addressing)
NR	Network Route (NETROUTE)
NSAP	Network Service Access Point (OSI addressing)
NSDU	Network Service Data Unit (OSI addressing)
NSM	Network Status Monitor
OCS	Open Communication Subsystem
OPEN7	OPEN GCOS 7 towards TCP/IP communications protocols
OSF	Open Systems Facility
OSI	Open Systems Interconnection
PAD	Packet Assembly Deassembly
PDN	Public Data Network
PDU	OSI layer Protocol Data Unit
PID	(French acronym for IDP)
PIDa	Addressing PID (DSA to ISO migration)
PIDp	Protocolar PID (DSA to ISO migration)
PLW	Pluri-Lingual West (Latin character set)

Acronym	Meaning
PPC	Program to Program Communication
PPCADU	Presentation Protocol Connect and Accept Data Unit (OSI addressing)
PPDU	Presentation Protocol Data Unit
PSI	Peripheral Subsystem Interface
PSAP	Presentation Service Access Point (OSI addressing)
PSDU	Presentation Service Data Unit (OSI addressing)
PSEL	Presentation layer SELector (OSI addressing)
PVE	Polled VIP Emulator
QD	Queue Definition
QMON	Queue MONitor
QOS	Quality Of Service (OSI addressing)
RAEH	Remote Administrative Exchange Handler (DSAC)
RARP	Reverse Address Resolution Protocol (TCP/IP)
RCT	Remote ConTroller (RMTCTLR)
RFA	Remote File Access (DSA file transfer protocol for UFT and DJP)
RIM	Request Initialization Mode
RIP	Routing Information Protocol
RNIS	(French acronym for ISDN)
RPL	Remote Physical Link (RMTPLINK)
RSC	Remote Session Control (RMTSESS)
RSP	Response, DSAC concept
RSYS	Remote System (RMTSYSTEM)
RTS	Remote Transport station (RMTTRANSPORT)

Acronym	Meaning
SA	ISO Addressing Subdomain (SDOM)
SAP	layer Service Access Point (OSI addressing)
SB	Statistics Blocks (network administration)
SC	Session Control
SCID	Session Control Identifier
SDP	Standard Device Protocol (presentation layer)
SDU	Service Data Unit (OSI addressing)
SEL	layer SELector (OSI addressing)
SID	ISO/DSA Specification
SIM	Set Initialization Mode
SIU	System Integration Unit
SNA	System Network Architecture
SNPA	SubNetwork Point of Attachment (OSI addressing)
SNMP	Simple Network Management Protocol
SPA	Service Processor Ares, applicable to DPS 7000
SPCADU	Session Protocol Connect and Accept Data Unit (OSI addressing)
SPDU	Session Protocol Data Unit (OSI addressing)
SR	Session Route (SESSROUTE)
SRST	System Resource and Status Table
SSAP	Session Service Access Point
SSDU	Session Service Data Unit (OSI addressing)
SSEL	Session layer SELector (OSI addressing)
SSn	Single Shift (terminal control character)
STID	(french acronym for DIWS)
SVR	Server

Acronym	Meaning
SYS	System
SYSGEN	System Generation for DNS and CNS 7
TCB	Text Control Block (MAM)
TCORR	Telecommunication CORRespondent dictionary
TCP	Transmission Control Protocol (UNIX transport)
TCRCDU	Transport Connection Request and Confirm Data Unit (OSI addressing)
TDP	transaction Dialog Protocol
TDS	Transactional Driven Subsystem
TM	Terminal Manager
TNS	Transport and Network Subsystem (comms server)
TP	1.Transport Protocol (TPROTOCOL) 2.Transaction Program (XCP2 or TDS environment)
TPDU	Transport Protocol Data Unit (ISO/DSA addressing)
TPI	Transport Programmatic Interface
TP-NODE	Transaction Program node (XCP2)
TRAILn	TRAILing TCP/IP headers protocol (TCP/IP)
TSAP	Transport Service Access Point (OSI addressing)
TSDU	Transport Service Data Unit (OSI addressing)
TSEL	Transport layer SELector (OSI addressing)
TSI	Transport Session Interface
TSYS	1.Telecommunication remote SYStem dictionary 2.Terminate SYStem main operator command
TTY	TeleTYpe (asynchronous terminal class)
TWA	Two Ways Alternate
TWS	Two Ways Simultaneous

Acronym	Meaning
UDP	User Datagram Protocol
UFT	Unified File Transfer, file transfer using DSA protocol RFA
UM	Unsolicited Message (DSAC)
URP	Unit Record Processor (V2)
UT	User of Transport (OSI addressing)
VCAM	Virtual Communications Access Method
	(session communication server)
VIP	Visual Information Projection (synchronous line procedure or synchronous terminal class)
WAN	Wide Area Network
XBF	eXtended BuFfer pool (for communications servers)
XCP	Extended Cooperative Protocol
XOPEN	Open to UNIX facilities
XTA	eXtended Twin Architecture
XTI	XOPEN Transport Interface

Index

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

S

T

U

V

W

X

Technical publication remarks form

Title :	DPS7000/XTA NOVASCALE 7000 Network Generation User's Guide Communications: FCP7

Reference N° :	47 A2 93UC 04	**Date:**	**August 2002**

ERRORS IN PUBLICATION

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

Your comments will be promptly investigated by qualified technical personnel and action will be taken as required.
If you require a written reply, please include your complete mailing address below.

NAME : ______________________________ Date : __________

COMPANY : ______________________________

ADDRESS : ______________________________

____

Please give this technical publication remarks form to your BULL representative or mail to:

Bull - Documentation Dept.
1 Rue de Provence
BP 208
38432 ECHIROLLES CEDEX
FRANCE
info@frec.bull.fr

Technical publications ordering form

To order additional publications, please fill in a copy of this form and send it via mail to:

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

Phone: +33 (0) 2 41 73 72 66
FAX: +33 (0) 2 41 73 70 66
E-Mail: srv.Duplicopy@bull.net

CEDOC Reference #	Designation	Qty
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
_ _ _ _ _ _ _ _ _ [_ _]		
[_ _] : The latest revision will be provided if no revision number is given.		

NAME: ______________________________ Date: __________

COMPANY: ______________________________

ADDRESS: ______________________________

____

PHONE: ______________________ FAX: ______________

E-MAIL: ______________________________

For Bull Subsidiaries:

Identification: ______________________________

For Bull Affiliated Customers:

Customer Code: ______________________________

For Bull Internal Customers:

Budgetary Section: ______________________________

For Others: Please ask your Bull representative.

BULL CEDOC
357 AVENUE PATTON
B.P.20845
49008 ANGERS CEDEX 01
FRANCE

REFERENCE
47 A2 93UC 04