GCOS 7 XTI

Named Service Manual

DPS7000/XTA
NOVASCALE 7000

Communications: GXTI

REFERENCE
47 A2 69UC 00

DPS7000/XTA NOVASCALE 7000

GCOS 7 XTI

Named Service Manual

Communications: GXTI

February 1995

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

REFERENCE
47 A2 69UC 00

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

MANUAL OBJECTIVES

This manual is designed to provide a complete description of the Name Services GXTI functions and primitives. The Name Services functions allow GCOS 7 XTI users to know and use the names of applications and ignore their actual addresses. This routine is to be used in addition to other XTI calls as defined in the XTI (X/Open Transport Interface) specification, but is not part of the XTI standard.

INTENDED AUDIENCE

Persons wanting to use or support GCOS 7 XTI Name Services :

STRUCTURE OF THE DOCUMENT

Chapter 1 is a brief overview.

Chapter 2 is a description of the user trace.

Chapter 3 is a description of the primitives.

Chapter 4 is a description of the database files.

A Glossary and Index then follow.

ASSOCIATED DOCUMENTS

The following publications of the DPS 7000 manual set should also be referred to:

GCOS 7 XTI User's Guide *47 A2 64UC*

VCAM-ISO User's Guide..... *47 A2 62UC*

VCAM-ISO Reference Manual (Part 1)..... *47 A2 60UC*

VCAM-ISO Reference Manual (Part 2)..... *47 A2 62UC*

VCAM-ISO Reference Manual (Part 3)..... *47 A2 63UC*

C System Primitives.....*47 A2 62 UL*

Networks Overview *47 A2 92UC*

Networks Generation *47 A2 93UC*

Networks User's Guide *47 A2 94UC*

TCP/IP 7 User's Guide..... *47 A2 30US*

OPEN 7 Administrator's Reference Manual..... *47 A2 31US*

OPEN 7 Administrator's Guide..... *47 A2 32US*

Table of Contents

Illustrations

Tables

1. OVERVIEW

1.1 BRIEF DESCRIPTION

The GXTI Name Service provides functions which, given an input application name (and endsystem name), return local (or remote) address. Given an input local (or remote) address, other functions return application name (and endsystem name).

It enables the user to know the application name and ignore its address.

The GXTI Name Service's functions may be used for the call to the GXTI's functions or to analyse their returns.

1.2 EXTERNAL FUNCTIONAL DESCRIPTION - CONCEPTS AND TERMINOLOGY

The GXTI Name Service is based on the functional descriptions of XTI Name Service of the *Guide & Reference Manual Bull DPX & DPX/2*, (86 A2 99SW) and *Guide & Reference Manual Bull DPX/20 (86 A2 51WE).*

1.2.1 Database Files

To make the relation between symbolic names and addresses, the GXTI Name Service's functions use files that contain descriptions of services and hosts :

- the OPEN 7 files /etc/services and /etc/hosts for TCP or UDP transport provider, when transport is provided by OPEN 7
- the GC0S 7 files SYS.DSACONF..ETC_SERVICES and SYS.DSACONF..ETC_HOSTS for TCP or UDP transport provider, when transport is provided by FCP 7/OCS
- the GCOS 7 files SYS.DSACONF..ETC_SERVICES and SYS.DSACONF..ETC_HOSTS for ISO, connection-oriented transport provider.

1.2.2 Error Handling

Each function has one or more error returns. Failures are indicated by a return value of -1.

t_errno, which is defined as a static integer in the header <gxti_ns.h>, holds the specific error number when such a failure occurs. This value is set when errors occur and must be tested only after an error has been indicated.

2. USER TRACE

The GXTI user trace allows the user to debug his own application. It gives a history of GXTI Name service invocations.

It has the same characteristics and the same commands as the VCAM user trace.

There are the three following commands:

`SVTR (START_VCAM_TRACE)`

to start the trace. It consists of opening a file and recording trace information. There is only one file per submitter.

`TVTR (TERMINATE_VCAM_TRACE)`

to stop the data collection and to close the file.

`DVTR (DISPLAY VCAM_TRACE)`

to display the trace parameters (How many users, which filters and levels values).

The information traced (at level 4) for each application's functions before and after calling the primitive are:

Table 2-1. Trace Parameters

	BEFORE	AFTER
T_GETLADDR	tp.tp_id tappl addr	tp.tp_id tappl addr
T_GETLNAME	tp.tp_id addr tappl	tp.tp_id addr tappl
T_GETRADDR	tp.tp_id endsys tappl addr	tp.tp_id endsys tappl addr
T_GETRNAME	tp.tp_id addr endsys tappl	tp.tp_id addr endsys tappl
T_GETTP	tp.tp_id	tp.tp_id tp.tp_name

The trace scanner allows you to edit and analyze the user trace previously collected. For example of trace output, see the GXTI User guide.

Refer to the *VCAM-ISO Reference Manual* for the trace scanner commands.

Refer to *GXTI User's Guide* and *VCAM-ISO Reference Manuals (3 volumes)* to get more information about trace functionalities.

3. DESCRIPTION OF PRIMITIVES

LEGEND

For each GXTI Name Service function, a table is given which summarizes the contents of the input or output parameters.

x

the parameter value is meaningful (input parameter must be set before the call and output parameter may be read after the call).

(x)

the content of the object pointed to by the x pointer is meaningful.

?

the parameter value is meaningful but the parameter is optional.

(?)

the content of the object pointed to by the ? pointer is optional.

/

the parameter value is meaningless.

3.1 T_GETLADDR

3.1.1 Name

t_getladdr - Get local address

3.1.2 Synopsis

```
#include <gxti_ns.h>
t_getladdr(tp, tappl, addr)
struct xtitp *tp;
char *tappl;
struct netbuf *addr;
```

3.1.3 Description

Given an input transport provider specified via tp (as returned by t_gettp()) and a transport application name in tappl, stored as a null-terminated C-string, t_getladdr() returns an address in the netbuf referenced by addr which may be used in a call to t_bind().

Table 3-1 Description of Primitives

Parameters	Before call	After call
tp->tp_id	x	/
tp->tp_name	/	/
tappl	x	/
addr->maxlen	x	/
addr->len	/	x
addr->buf	x	(x)

3.1.4 Return Value

Upon successful completion a value of 0 is returned. Otherwise a value of -1 is returned and t_errno is set to indicate an error.

3.1.5 Errors

On failure t_errno is set to one of the following:

[TAPPLNOTFOUND]	The transport application specified by tappl is not defined in the service database.
[TBADBUF]	Illegal buffer description.
[TBUFOVFLW]	The address to be returned in addr is larger than the size specified in the maxlen field of the addr netbuf structure.
[TDBASENAV]	database not available.
[TNOPROVRESP]	The transport provider is not responding.
[TNOTSUPPORT]	The tp parameter does not refer to a supported Transport Provider.
[TPROVFAIL]	The transport provider processing failed.
[TSYSERR]	A system error has occurred during execution of this function.

3.1.6 Files

SYS.DSACONF..ETC_SERVICES (GCOS 7 file)

/etc/services (OPEN 7 file)

3.1.7 See Also

See Also: t_getlname(), t_getraddr(), t_getrname(), t_gettp().

3.2 T_GETLNAME

3.2.1 Name

t_getlname - Parse local address

3.2.2 Synopsis

```
#include <gxti_ns.h>
t_getlname(tp, addr, tappl)
struct xtitp *tp;
struct netbuf *addr;
struct netbuf *tappl;
```

3.2.3 Description

Given an input local address in the netbuf referenced by addr (as returned by the t_bind() call), and in the format defined for the transport provider specified by tp (as returned by t_gettp()), t_getlname() parses the address and looks up in the database the transport application name referenced by the address.

It returns the name string in the tappl input structure.

The length of the netbuf includes the NULL (0) byte which terminates a C-string.

If the application name is not in the database, the string returned in tappl represents

- the port number (in decimal format) for TCP or UDP transport provider,
- the TSEL (in hexadecimal format) for ISO transport provider.

In this case, -1 will be returned and t_errno will be set to [TAPPLNOTFOUND].

When no error is returned, the string returned in tappl is usable as input to the t_getladdr() call.

Table 3-2. Summary of Input/Output Parameters for T_GETLNAME

Parameters	Before call	After call
tp->tp_id	x	/
tp->tp_name	/	/
addr->maxlen	/	/
addr->len	x	/
addr->buf	(x)	/
tappl->maxlen	x	/
tappl->len	/	x
tappl->buf	x	(x)

3.2.4 Return Value

Upon successful completion a value of 0 is returned. Otherwise a value of -1 is returned and t_errno is set to indicate an error.

3.2.5 Errors

On failure t_errno is set to one of the following:

[TAPPLNOTFOUND]	The transport application specified by tappl is not defined in the service database.
[TBADADDR]	The address in addr is not in a recognizable format.
[TBADBUF]	Illegal buffer description.
[TBUFOVFLW]	The address to be returned in addr is larger than the size specified in the maxlen field of the addr netbuf structure.
[TDBASENAV]	database not available.
[TNOPROVRESP]	The transport provider is not responding.
[TNOTSUPPORT]	The tp parameter does not refer to a supported Transport Provider.
[TPROVFAIL]	The transport provider processing failed.
[TSYSERR]	A system error has occurred during execution of this function.

3.2.6 Files

SYS.DSACONF..ETC_SERVICES (GCOS 7 file)

/etc/services (OPEN 7 file)

3.2.7 See Also

See Also: t_getladdr(), t_getraddr(), t_getrname(), t_gettp().

3.3 T_GETRADDR

3.3.1 Name

t_getraddr - Get remote address

3.3.2 Synopsis

```
#include <gxti_ns.h>
t_getraddr(tp, endsys, tappl, addr)
struct xtitp *tp;
char *endsys;
char *tappl;
struct netbuf *addr;
```

3.3.3 Description

Given an input endsystem name (endsys) and transport application name (tappl), both stored as null-terminated C-strings, t_getraddr() returns an address in the netbuf referenced by addr which may be used in a call to t_connect() or t_sndudata() to access the specified transport application on the specified endsystem.

Notice that when endsys is the local hostname (as returned by h_sysid() system primitive), the returned address is not built from hosts file information, but contains

- IP address zero for TCP or UDP transport provider,
- no NSAP for ISO transport provider.

Notice that to avoid as much as possible a TBADADDR error for too long address reason at t_connect() call:

With ISO transport provider, when the host line in hosts file contains several NSAPs, the returned address contains the whole NSAP list if the total address is not longer than 56 bytes (witch is the maximum length of address at t_connect()). Otherwise, the returned address contains only the beginning of the list, and the remainding NSAPs are ignored.

Table 3-3. Summary of Input/Output Parameters for T_GETRADDR

Parameters	Before call	After call
tp->tp_id	x	/
tp->tp_name	/	/
endsys	x	/
tappl	x	/
addr->maxlen	x	/
addr->len	/	x
addr->buf	x	(x)

3.3.4 Return Value

Upon successful completion a value of 0 is returned. Otherwise a value of -1 is returned and t_errno is set to indicate an error.

3.3.5 Errors

On failure t_errno is set to one of the following:

[TAPPLNOTFOUND]	The transport application specified by tappl is not defined in the service database.
[TBADBUF]	Illegal buffer description.
[TBUFOVFLW]	The address to be returned in addr is larger than the size specified in the maxlen field of the addr netbuf structure.
[TDBASENAV]	database not available.
[TENDSYSNOTFOUND]	The endsystem specified by endsys is not defined in the endsystem database.
[TNOPROVRESP]	The transport provider is not responding.
[TNOTSUPPORT]	The tp parameter does not refer to a supported Transport Provider.
[TPROVFAIL]	The transport provider processing failed.
[TSYSERR]	A system error has occurred during execution of this function.

3.3.6 Files

SYS.DSACONF..ETC_HOSTS (GCOS 7 file)

SYS.DSACONF..ETC_SERVICES (GCOS 7 file)

/etc/hosts (OPEN 7 file)

/etc/services (OPEN 7 file)

3.3.7 See Also

See Also: t_getladdr(), t_getlname(), t_getrname(), t_gettp().

3.4 T_GETRNAME

3.4.1 Name

t_getrname - Parse remote address

3.4.2 Synopsis

```
#include <gxti_ns.h>
t_getrname(tp, addr, endsys, tappl)
struct xtitp *tp;
struct netbuf *addr;
struct netbuf *endsys;
struct netbuf *tappl;
```

3.4.3 Description

Given an input remote address in the netbuf referenced by addr, as returned by the t_listen() and t_accept() calls, and in the format defined for the transport provider specified by tp (as returned by t_gettp()), t_getrname() parses the address and looks up in the database the endsystem name and the transport application name referenced by the address.

It returns the name strings in endsys and tappl, which each must point to netbuf structures.

The length of the netbuf includes the NULL (0) byte which terminates a C-string.

If the endsystem name is not in the database, the string returned in endsys represents

- the IP address (in decimal dot notation xx.xx.xx.xx) for TCP or UDP transport provider,
- the NSAP (in hexadecimal format) for ISO transport provider.

If the application name is not in the database, the string returned in tappl represents

- the port number (in decimal format) for TCP or UDP transport provider
- the TSEL (in hexadecimal format) for ISO transport provider.

Then -1 will be returned and t_errno will be set to the appropriate code [TAPPLNOTFOUND], [TENDSYSNOTFOUND] or [TBADNAME].

When no error is returned, the strings returned in endsys and tappl are usable as input to the t_getraddr() call.

Notice that when addr contains

- IP address zero for TCP or UDP transport provider,
- no NSAP for ISO transport provider,

the returned endsys name is not retrieved from hosts file, but is the local hostname (as returned by h_sysid() system primitive).

Table 3-4. Summary of Input/Output Parameters for T_GETRNAME

Parameters	Before call	After call
tp->tp_id	x	/
tp->tp_name	/	/
addr->maxlen	x	/
addr->len	/	/
addr->buf	(x)	/
endsys->maxlen	x	/
endsys->len	/	x
endsys->buf	x	(x)
tappl->maxlen	x	/
tappl->len	/	x
tappl->buf	x	(x)

3.4.4 Return Value

Upon successful completion a value of 0 is returned. Otherwise a value of -1 is returned and t_errno is set to indicate an error.

3.4.5 Errors

On failure t_errno is set to one of the following:

[TAPPLNOTFOUND]	The transport application specified by tappl is not defined in the service database.
[TBADBUF]	Illegal buffer description.
[TBADNAME]	The transport application specified by tappl is not defined in the service database and the endsystem specified by endsys is not defined in the endsystem database.
[TBUFOVFLW]	The address to be returned in addr is larger than the size specified in the maxlen field of the addr netbuf structure.
[TDBASENAV]	database not available.

[TENDSYSNOTFOUND]	The endsystem specified by endsys is not defined in the endsystem database.
[TNOPROVRESP]	The transport provider is not responding.
[TNOTSUPPORT]	The tp parameter does not refer to a supported Transport Provider.
[TPROVFAIL]	The transport provider processing failed.
[TSYSERR]	A system error has occurred during execution of this function.

3.4.6 Files

SYS.DSACONF..ETC_HOSTS (GCOS 7 file)

SYS.DSACONF..ETC_SERVICES (GCOS 7 file)

/etc/hosts (OPEN 7 file)

/etc/services (OPEN 7 file)

3.4.7 See Also

See Also: t_getladdr(), t_getlname(), t_getraddr(), t_gettp().

3.5 T_GETTP

3.5.1 Name

t_gettp - Get transport provider

3.5.2 Synopsis

```
#include <gxti_ns.h>
t_gettp(tp)
struct xtitp *tp;
```

3.5.3 Description

Given an input transport provider id code in tp->tp_id (TPID_xxx from gxti_ns.h), t_gettp() completes the tp structure and in particular returns a string in tp->tp_name usable in a call to t_open() as a transport provider id.

Table 3-5. Summary of Input/Output Parameters for T_GETTP

Parameters	Before call	After call
tp->tp_id	x	/
tp->tp_name	/	(x)

3.5.4 Return Value

Upon successful completion a value of 0 is returned. Otherwise a value of -1 is returned and t_errno is set to indicate an error.

3.5.5 Errors

On failure t_errno is set to one of the following:

[TBADBUF] Illegal buffer description.

[TNOTSUPPORT]	The tp parameter does not refer to a supported Transport Provider.
[TSYSERR]	A system error has occurred during execution of this function.

3.5.6 See Also

SEE ALSO: t_getladdr(), t_getlname(), t_getraddr(), t_getrname().

3.6 THE <GXTI_NS.H> HEADER

gxti_ns.h contains the definition of the GXTI Name Service primitives, and also the following information:

```
/*-------------------------------------------------*/
/* transport providers ids                         */
/*-------------------------------------------------*/
#define TPID_OSI_COTS  0  /* OSI connection oriented provider */
#define TPID_OSI_CLTS  1  /* OSI connectionless provider */
#define TPID_TCP       2  /* TCP provider */
#define TPID_UDP       3  /* UDP provider */

/*-----------------------------------------------          ----*/
/* Proprietary transport ids may be added here      */
/* beginning with TPID_FIRST_PROP_ID                  */
/*-----------------------------------------------          ----*/
#define TPID_FIRST_PROP_ID   100

/*-------------------------------------------------*/
/* new errors                                       */
/*-------------------------------------------------*/
#define TENDSYSNOTFOUND  100  /* No endsystem in endsystem
database */
#define TAPPLNOTFOUND    101  /* No trans. application in appl db
*/
#define TBADBUF          150  /* illegal buffer description */
#define TDBASENAV        151  /* database not available */

/*-----------------------------------------------     -*/
/* GXTI Transport provider structure :              */
/*                      input to GXTI naming calls.       */
/*-----------------------------------------------    */
struct xtitp {
    int    tp_id;       /* provider ID (TPID_xxx values above) */
    char   *tp_name;    /* provider name */
};
```

4. DESCRIPTION OF DATABASE FILES

4.1 GCOS 7 HOSTS FILE

4.1.1 Name

SYS.DSACONF..ETC_HOSTS - GCOS 7 host name database

4.1.2 Description

The hosts file contains information regarding the known hosts on the network.

For each host, a single line should be present with the following information:

- address
- official host name
- aliases

Items are separated by any numbers of blanks characters.

A "#" indicates the beginning of a comment; characters up to the end of the line are not interpreted by routines which search the file.

For TCP or UDP transport provider, address is the IP network address (in 4 parts decimal dot notation, see example below).

For ISO transport provider, address is a NSAP list (in hexadecimal format, prefixed by "@iso," , see example below).

Host names (and aliases) may contain any printable character other than a field delimiter, newline or comment character.

4.1.3 Application Usage

This file is used by the primitives defined in the GXTI Name Service, for

- ISO transport provider,
- TCP or UDP transport provider, when transport is provided by FCP 7/OCS.

4.1.4 Example

```
# ETC_HOSTS file
# --------------
#
# Internet address examples:
#
129.182.51.8    host1 HOST1  # IP addr: 81b63308 hexa
# The previous line is an internet host description
# that contains:
#   - the internet address [129.182.51.8], 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 access point examples:
#
@iso,3802c4d6   host1 host3 HOST3 host_3 HOST_3
# The previous line is an ISO host description
# that contains:
#   - a 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
```

4.2 GCOS 7 SERVICES FILE

4.2.1 Name

SYS.DSACONF..ETC_SERVICES - GCOS 7 service name database

4.2.2 Description

The services file contains information regarding the known services, identified bye its address, available in the the network.

For each service, a single line should be present with the following information:

```
- offical service name
- address
- transport provider name
- aliases
```

Items are separated by any numbers of blanks characters.

A "#" indicates the beginning of a comment; characters up to the end of the line are not interpreted by routines which search the file.

The address and transport provider name are considered as a single item; a "/" is used to separate the address and the transport provider name (see example below).

For TCP or UDP transport provider, address is the port number (in decimal format, see example below).

For ISO transport provider, address is the TSEL (in hexadecimal format, see example below).

Host names (and aliases) may contain any printable character other than a field delimiter, newline or comment character.

4.2.3 Application Usage

This file is used by the primitives defined in the GXTI Name Service, for

- ISO transport provider,
- TCP or UDP transport provider, when transport is provided by FCP 7/OCS.

4.2.4 Example

```
# ETC_SERVICES file
# -----------------
#
# tcp service example:
#
service1    65521/tcp SERVICE1    # port: fff1 hexa
# The previous line is a TCP service description that
# contains:
#   - the service name  [service1]
#   - the port number   [65521]
#   - the protocol      [tcp]
#   - an alias          [SERVICE1]
#   - a comment         [# port: fff1 hexa]
#
ftp-data    20/tcp
ftp         21/tcp
telnet      23/tcp
smtp        25/tcp  mail
#
# udp service example:
#
service2    65522/udp             # port: fff2 hexa
# The previous line is an UDP service description that
# contains:
#   - the service name  [service2]
#   - the port number   [65522]
#   - the protocol      [udp]
#   - no alias
#   - a comment         [# port: fff2 hexa]
#
who         513/udp whod
router      520/udp route routed
#
# iso service example:
#
service3    1a050f85/isoco        # tsel: 1a050f85 hexa
# The previous line is an ISO,connection-oriented service
# description that contains:
#   - the service name       [service3]
#   - the transport selector [1a050f85]
#   - the protocol           [isoco]
#   - no alias
#   - a comment         [# tsel: 1a050f85 hexa]
```

4.3 OPEN 7 HOSTS FILE

4.3.1 Name

/etc/hosts - OPEN 7 host name database

4.3.2 Description

The hosts file contains information regarding the known hosts on the network.

For each host, a single line should be present with the following information:

```
- address
- official host name
- aliases
```

Items are separated by any numbers of blanks and/or tab characters.

A "#" indicates the beginning of a comment; characters up to the end of the line are not interpreted by routines which search the file.

Address is the IP network address (in dot notation, see example below).

Host names (and aliases) may contain any printable character other than a field delimiter, newline or comment character.

4.3.3 Application Usage

This file is used by the primitives defined in the GXTI Name Service, for TCP or UDP transport provider, when transport is provided by OPEN 7.

4.3.4 Example

```
# /etc/hosts
# ----------
#
# localhost: loopback driver
127.0.0.1      localhost local
#
129.182.51.40  bach
129.182.51.5   ravel
```

4.4 OPEN 7 SERVICES FILE

4.4.1 Name

/etc/services - OPEN 7 service name database

4.4.2 Description

The services file contains information regarding the known services, identified bye its address, available in the the network.

For each service, a single line should be present with the following information:

```
offical service name
address
transport provider name
aliases
```

Items are separated by any numbers of blanks and/or tab characters.

A "#" indicates the beginning of a comment; characters up to the end of the line are not interpreted by routines which search the file.

The address and transport provider name are considered as a single item; a "/" is used to separate the address and the transport provider name (see example below).

Address is the port number (in decimal format, see example below).

Host names (and aliases) may contain any printable character other than a field delimiter, newline or comment character.

4.4.3 Application Usage

This file is used by the primitives defined in the GXTI Name Service, for TCP or UDP transport provider, when transport is provided by OPEN 7.

4.4.4 Example

```
# /etc/services
# -------------
#
ftp-data    20/tcp
ftp         21/tcp
telnet      23/tcp
smtp        25/tcp  mail
#
# UNIX specific sevices;
# these are NOT officially assigned
#
login       513/tcp
shell       514/tcp cmd
who         513/udp whod
router      520/udp route routed
uucp        540/tcp uucpd
```

Glossary

FCP7	FDDI Communication Processor 7
GCOS	General Comprehensive Operating System
GXTI	GCOS7 X/Open-like Transport Interface
IP	Internet Protocol
ISO	International Standard Organization
NSAP	Network Service Access point
OCS	Open Communication Subsystem
OSI	Open System Interconnection
TCP	Transmission Control Protocol
Transport provider	Transport protocol that provides the service of the Transport layer
TSAP	Transport Service Access Point
TSEL	Transport Selector
UDP	User Datagram Protocol
XPG	X/Open Portability Guide

Index

Technical publication remarks form

Title :	DPS7000/XTA NOVASCALE 7000 GCOS 7 XTI Named Service Manual Communications: GXTI

Reference N° :	47 A2 69UC 00	**Date:**	**February 1995**

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 69UC 00