



**DT-6061**

**Embedded Network Processor  
Platform**

**User's Manual**



**ISSUE 2**

**RELEASE VERSION 16.0**

721 Route 202-206  
Bridgewater, NJ 08807  
fax: 908.218.1736  
phone: 908.218.0500  
email: [sales@datatekcorp.com](mailto:sales@datatekcorp.com)  
<http://www.datatekcorp.com>



## Table of Contents

<b>TABLE OF CONTENTS</b> .....	<b>2</b>
<b>IMPORTANT SAFETY INSTRUCTIONS</b> .....	<b>7</b>
<b>1 INTRODUCTION</b> .....	<b>8</b>
1.1 MULTIPLE APPLICATION SUPPORT.....	8
1.2 HIGH AVAILABILITY OPTION .....	10
<b>2 PHYSICAL DESCRIPTION</b> .....	<b>11</b>
2.1 POWER INTERFACES.....	11
2.1.1 DC Power.....	11
2.1.2 AC Power.....	11
2.2 ALARM .....	11
2.3 CON (CONSOLE) .....	11
2.4 RS-232 / V.35.....	11
2.5 10BASE-T LAN.....	12
2.6 DSU .....	12
2.7 LEDs.....	12
<b>3 INSTALLATION</b> .....	<b>13</b>
3.1 EQUIPMENT.....	13
3.1.1 Power Reconfiguration Steps for 115V AC (-5V DC) or -48V DC Operation.....	13
3.1.2 Mounting and Wiring .....	13
3.1.2.1 Stand-Alone DT-6061 ( <i>AC to -5V DC Power Input</i> ).....	14
3.1.2.2 Rack-Mount DT-6061 ( <i>AC to -5V DC Power Input</i> ).....	14
3.1.2.3 Stand-Alone DT-6061 ( <i>48V DC Power Input</i> ).....	14
3.1.2.4 Rack-Mount DT-6061 ( <i>48V DC Power Input</i> ).....	15
<b>4 CABLING</b> .....	<b>16</b>
4.1 SERIAL CONSOLE SETUP .....	16
4.2 CONFIGURATION NOTES .....	17
<b>5 QUICK START CONFIGURATION GUIDE</b> .....	<b>18</b>
5.1 PLATFORM CONFIGURATION.....	18
5.2 PLATFORM SOFTWARE REGISTRATION.....	18



5.3	APPLICATION INSTALLATION.....	18
5.4	APPLICATION SOFTWARE REGISTRATION .....	19
5.5	ASSIGNMENT OF AN APPLICATION TO AN INSTANCE .....	19
6	PLATFORM COMMANDS .....	20
6.1	INPUT CONVENTIONS .....	20
6.2	LOGIN.....	20
6.3	LOGOUT .....	21
6.4	DATE AND TIMEZONE .....	21
6.5	CHANGE PASSWORD .....	22
6.6	RESET PASSWORD .....	22
6.7	SYSTEM PARAMETER CONFIGURATION .....	22
6.8	SET CONSOLE TIMEOUT.....	23
6.9	SYSTEM LABEL CONFIGURATION.....	23
6.10	CONFIGURE THE BANNER.....	23
6.11	REBOOTING THE DT-6061.....	23
6.12	SYSTEM CONSOLE USER HELP .....	24
6.13	CONFIGURE HOST NAMES.....	24
6.14	CONFIGURE DNS (DOMAIN NAME SERVER).....	24
6.15	OPERATIONAL APPLICATIONS CONFIGURATION .....	25
6.16	RESTART AN INSTANCE .....	25
6.17	REMOVE OR RESTORE AN INSTANCE .....	25
6.18	RESET AN APPLICATION PASSWORD .....	26
6.19	MANAGE LIST OF COMPANION DT PRODUCTS.....	26
6.20	CONFIGURE SNMP .....	27
6.21	CLOSED USER GROUPS .....	27
6.22	VERIFY CONFIGURATION SETTINGS - VFY.....	28
6.22.1	Verify Configured Application Instances.....	28
6.22.2	Verify Running Applications .....	28
6.22.3	Verify Module .....	29
6.23	DISPLAY CONNECTIONS .....	29
6.24	DISPLAY MEASUREMENTS.....	30



6.25	BACKUP AND RETRIEVE APPLICATION CONFIGURATION DATA .....	31
7	DIAGNOSTICS .....	32
7.1	PING.....	32
7.2	TRTE ( OR TRACERT) .....	32
7.3	DIAG COMMAND .....	32
7.3.1	address .....	32
7.3.2	cpu .....	32
7.3.3	dev .....	33
7.3.4	ping.....	34
7.3.5	tracert.....	34
8	SNMP .....	35
8.1	DT-6061 SNMP MIB VARIABLE DATABASE .....	35
9	APPLICATION AND PLATFORM SOFTWARE INSTALLATION & UPGRADE .....	36
9.1	INSTALL AN APPLICATION .....	36
9.2	APPLICATION SOFTWARE REGISTRATION .....	36
9.3	UNINSTALL AN APPLICATION .....	37
9.4	INSTALLING A PLATFORM UPGRADE .....	37
9.5	PLATFORM SOFTWARE REGISTRATION.....	37
10	HIGH AVAILABILITY OPTION .....	38
10.1	MODULE CONFIGURATION FOR HIGH AVAILABILITY .....	38
10.2	APPLICATION INSTALLATION AND OPERATIONAL CONFIGURATION .....	39
10.2.1	Application Installation.....	39
10.2.2	Operational Configuration.....	39
10.3	AUTOMATIC STANDBY CONFIGURATION UPDATE .....	40
10.4	SWITCHOVER .....	40
10.5	COMMANDS NOT AVAILABLE IN <i>STANDBY MODE</i> .....	40
11	APPENDIX A: BRIEF DESCRIPTIONS OF DT-6061 APPLICATIONS ....	42
11.1	IP-BHI .....	42
11.2	TN3270 .....	42
11.3	IP-B2APRT .....	42



11.4	IP-E2A .....	42
11.5	IP-FANOUT.....	42
11.6	IP-LTCN.....	42
11.7	X25PAD .....	43
11.8	IP-SLM.....	43
11.9	IP-SPOOL.....	43
11.10	IP-VCON .....	44
11.11	ONSITE.....	44
12	APPENDIX B: CONSOLE CABLE WIRING DIAGRAMS .....	45
12.1	THE DB9 CONSOLE ADAPTER .....	46
13	APPENDIX C: TCP/UDP PORT NUMBERS.....	47
14	APPENDIX D: THE DISPLAY MEASUREMENTS (DM) REPORT .....	48
14.1	IP.....	48
14.2	TCP .....	51
14.3	UDP .....	54
14.4	ICMP.....	55
15	APPENDIX E: DT-6061 SPECIFICATIONS.....	57
15.1	CON (CONSOLE) .....	57
15.2	RS-232/V.35 .....	57
15.2.1	RS-232C .....	57
15.2.2	V.35 .....	57
15.3	10-BASE-T LAN.....	57
15.4	DSU .....	57
15.5	PHYSICAL DIMENSIONS.....	57
15.6	ENVIRONMENTAL OPERATING RANGE .....	57
15.7	POWER REQUIREMENTS.....	57
15.8	REGULATORY INFORMATION.....	58
15.8.1	DT-6061 Stand-Alone .....	58
15.8.2	FCC Part 68 Information .....	58
15.8.3	Industry Canada CS03 Certification Information .....	58
15.8.4	NEBS.....	59



**16 HARDWARE WARRANTY .....60**

**17 END-USER LICENSE AGREEMENT FOR SOFTWARE .....60**

**17.1 SOFTWARE LICENSE.....60**

**17.2 INTELLECTUAL PROPERTY RIGHTS.....60**

**17.3 SOFTWARE SUPPORT .....60**

**17.4 EXPORT RESTRICTIONS .....61**

**17.5 LIMITED WARRANTY .....61**

**17.6 NO OTHER WARRANTIES .....61**

**17.7 SPECIAL PROVISIONS .....61**

**18 LIMITATION OF LIABILITY.....61**



## Important Safety Instructions



The exclamation point within an equilateral triangle is intended to alert the user to the presence of important operating and maintenance (servicing) instructions in the literature accompanying the product.

When installing, operating, or maintaining this equipment, basic safety precautions should always be followed to reduce the risk of fire, electric shock, and injury to persons, including the following:

- Read and understand all instructions.
- Follow all warnings and instructions marked on this product.
- For information on proper mounting instructions, consult the User's Manual provided with this product.
- The telecommunications interface should not leave the building premises unless connected to telecommunication devices providing primary and secondary protection.
- This product should only be operated from the type of power source indicated in the User's Manual.
- This unit must be powered from either -48 V DC or AC voltage sources. See User's Manual before connecting to the power source.
- The -48 V DC input terminals are only provided for installations in Restricted Access Areas locations.
- Do not use this product near water, for example, in a wet basement.
- Never touch non-insulated wiring or terminals carrying direct current or leave this wiring exposed. Protect and tape wiring and terminals to avoid risk of fire, electric shock, and injury to service personnel.
- To reduce the risk of electrical shock, do not disassemble this product. Only trained personnel should perform servicing. Opening or removing covers and/or circuit boards may expose you to dangerous voltages or other risks. Incorrect re-assembly can cause electric shock when the unit is subsequently used.
- For a unit intended to be powered from -48 V DC voltage sources, read and understand the following:
  - This equipment must be provided with a readily accessible disconnect device as part of the building installation.
  - Ensure that there is no exposed wire when the input power cables are connected to the unit.
  - Installation must include an independent frame ground drop to building ground. Refer to User's Manual



This symbol is marked on the DT-6061, adjacent to the ground (earth) area for the connection of the ground (earth) conductor.

- This Equipment is to be Installed Only in Restricted Access Areas on Business and Customer Premises Applications in Accordance with Articles 110-16, 110-17, and 110-18 of the National Electrical Code, ANSI/NFPA No. 70. Other Installations Exempt from the Enforcement of the National Electrical Code May Be Engineered According to the Accepted Practices of the Local Telecommunications Utility.
- For a unit equipped with an AC Wall Plug-In Unit, read and understand the following:
  - The unit was tested with the K'TRON, Model KA-52A Wall Plug-In Unit.
  - Unplug this product from the wall outlet before cleaning. Do not use liquid cleaners or aerosol cleaners. Use a damp cloth for cleaning.
  - Do not staple or otherwise attach the power supply cord to the building surfaces.
  - Do not overload wall outlets and extension cords as this can result in the risk of fire or electric shock.
  - The socket outlet shall be installed near the equipment and shall be readily accessible.
  - The Wall Plug-In unit may be equipped with a three-wire grounding type plug, a plug having a third (grounding) pin. This plug is intended to fit only into a grounding type power outlet. Do not defeat the safety purpose of the grounding type plug.
  - Do not allow anything to rest on the power cord. Do not locate this product where persons walking on it may abuse the cord.
  - Unplug this product from the wall outlet and refer servicing to qualified service personnel under the following conditions:
    - a) When the power supply cord or plug is damaged or frayed.
    - b) If liquid has been spilled into the product.
    - c) If the product has been exposed to rain or water.
    - d) If the product does not operate normally by following the operating instructions. Adjust only those controls that are covered by the operating instructions because improper adjustment of other controls may result in damage and will often require extensive work by qualified technician to restore the product to normal operation...
    - e) If the product has been dropped or the cabinet has been damaged.
    - f) If the product exhibits a distinct change in performance.

---

## Save These Instructions

---

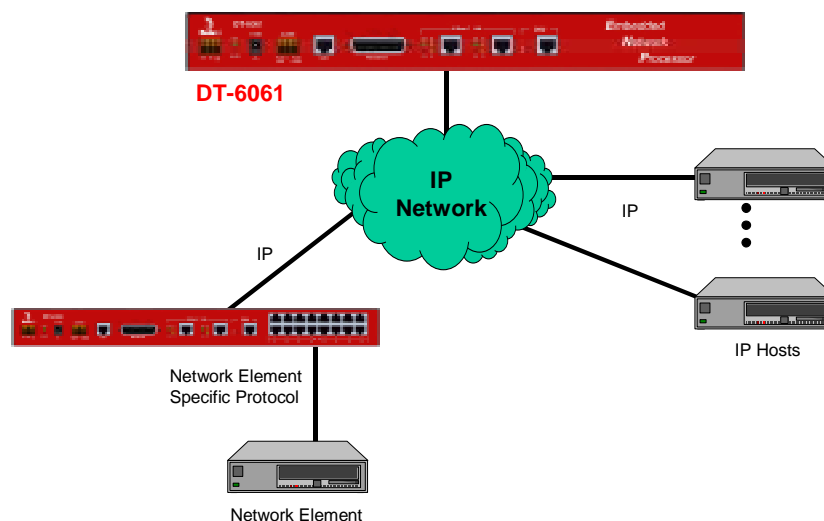


# 1 INTRODUCTION

The DT-6061 (*Embedded Network Processor*) is a network applications computer that applies customer specific protocol operations (*protocol translation and termination*) of network connections that require such a function in a distributed network.

The DT-6061 supports a multitude of applications, and may support combinations of these applications simultaneously. The elements common to all applications are provided by the DT-6061 hardware and operating environment. In addition, the DT-6061 has an SNMP agent that is completely independent of any application.

The basic configuration for the DT-6061 may be as follows:



In the above figure representing a generic configuration for a DT-6061, the Network Element (*NE*) communicates with IP Hosts over an IP Network using the DT-6061 as an intermediary for protocol conversion / termination. The NE uses its specific communications protocol to talk through a DT-4000 or some other device(s) that will bridge the NE with an IP infrastructure. The DT-6061 running one of its custom applications performs the necessary protocol translation or termination allowing the NE and IP-Hosts to ultimately communicate.

## 1.1 MULTIPLE APPLICATION SUPPORT

The DT-6061 can support one or more different types of applications and multiple **instances** of each application. An application **instance** can be described as a unit of configuration parameters as a specific DT-6061 application defines them. In other words, each **instance** of an application is a completely separate process where all aspects of the configuration and operation of the application are performed entirely within that process.





Currently the DT-6061 is factory configured to order to support 1 to 5 application types. The table below shows the list of the currently available applications.

Application Name	Description
IP-B2APRT	Allows a BiSync host to print to an ASCII printer located on an IP network
IP-BHI	Allows a legacy BiSync host to connect to an IP network
IP-E2A	Allows E2A telemetry ports of a network element to communicate over IP networks to OS hosts and eliminates the need for digital bridging
IP-FANOUT	Allows diverse LAN segments to be connected over a WAN as a cohesive IP network
IP-LTCN	Allows legacy telephone line test equipment to interface with IP-based OS hosts
IP-SLM	Allows specialized (B)X.25 telephone switch interfaces and other network elements to communicate over IP networks with OS hosts
IP-SPOOL	Eliminates underutilized physical printers by providing logical printers and print spooling per RFC1179 to a physical, shared print resource anywhere in an IP network
IP-VCON	Takes over the mundane task of monitoring a large and varied set of device consoles autonomously, collects console messages, and intervenes automatically when necessary according to user-defined scripts
ONSITE	Generates ASCII alarms based on network alarm grid open and closed contacts
TN3270	Allows a 3270 client on an IP network to access a BiSync host
X25PAD	Eliminates the need, complexity, and expense of specialized X.25 equipment in an IP network by allowing an X.25/BX.25 device, such as a network element's OS port, to be connected to a synchronous port anywhere in the IP network and communicate with a host via IP.

Additional description for each application is provided in Appendix A, section 11. In addition, the user manual for each application can be downloaded at no charge from the Datatek web site:

[http://www.datatekcorp.com/support/support\\_soft\\_firm\\_updates.html](http://www.datatekcorp.com/support/support_soft_firm_updates.html)



## 1.2 HIGH AVAILABILITY OPTION

The DT-6061 is available in a redundant configuration to provide high availability of the user's applications. A single *active* DT-6061 running a set of application **instances** is coupled with a *standby* DT-6061 that is ready to take over automatically if the *active* DT-6061 fails for any reason. After the failed DT-6061 recovers, it is returned to service as *standby*.



## 2 PHYSICAL DESCRIPTION



### 2.1 POWER INTERFACES

#### 2.1.1 DC POWER

The DT-6061 rack-mount or stand-alone accepts DC power input directly from a 48V DC power source and connects into the three position (accepting return, minus and ground, power wires) terminal block labeled 48V DC on the DT-6061 faceplate. The terminal block connectors accommodate 10 awg to 14 awg (American Wire Gauge) wire. A strain relief clamp is available separately for DC wire stabilization.

#### 2.1.2 AC POWER

For this application, a separate AC power supply is available. The power supply has a six-foot long cable that terminates with a barrel connector. The power supply plugs into a standard 115V AC outlet. The output of this supply is 5V DC. The barrel connector plugs into the circular connector labeled 5V DC on the DT-6061 faceplate.

### 2.2 ALARM

The Alarm Grid connector is a three position (Failed Open, Closed, Failed Closed) terminal block labeled ALARM on the DT-6061 faceplate. The terminal block connectors accommodate 10 awg to 14 awg (American Wire Gauge) wire.

### 2.3 CON (CONSOLE)

There are two types of consoles in a DT-6061: the single serial platform console and multiple telnet application consoles. The serial console is used for initial configuration of the platform and installation and control of applications. Each telnet console is used to configure the application database for one instance of an application.

The serial console port interface, which is labeled as CON on the DT-6061, requires a standard RJ45 terminated, twisted pair, data cable. It connects as data terminating equipment (DTE) to an asynchronous device and uses RS-232C signaling. Connection to the DT-6061 console is required for any DT-6061 platform administration. Otherwise, the console can be disconnected during normal operation and only the logical telnet console used to administer an instance of an application. See section 4.1 for details on cabling the serial console. The serial console is configured as 9600 bps, 8 bits, and no parity.

### 2.4 RS-232 / V.35

Through a DB25, RS-530 connector the DT-6061 supports two, software selectable, device interfaces (V.35 and RS-232C). ***This connector is currently slated for future use.***



## 2.5 10BASE-T LAN

These interfaces require a standard RJ45 terminated Category 5, twisted pair, data cable. They connect to a 10BaseT hub or Router on the local LAN segment.

## 2.6 DSU

The DT-6061 DSU interface uses an industry standard RJ48C connector. ***This connector is currently slated for future use.***

## 2.7 LEDs

The DT-6061 faceplate contains light emitting diodes (LEDs) used to report DT-6061 activity and behavior.

LED Function	LED Color	LED Description
PWR	Green	Unit Power Indicator
ALARM	Red	Reset Indicator & General Failure Indicator
LNK	Green	10 Base-T Link Indicator
COL	Red	10 Base-T Collision Indicator
RX	Yellow	10 Base-T Receive Packet Indicator
TX	Yellow	10 Base-T Transmit Packet Indicator
DSU	Red	LOS Indicator



### 3 INSTALLATION

This chapter contains the steps required to install and configure the DT-6061.

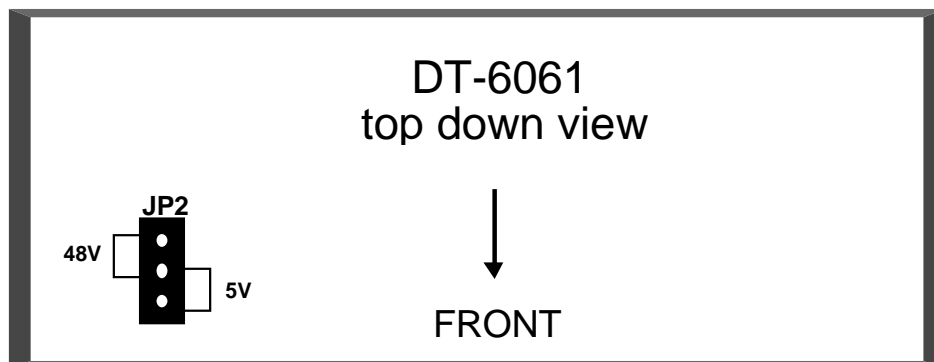


#### 3.1 EQUIPMENT

Unpack and inspect the DT-6061 unit and other components and have on hand a #2 Phillips and medium-sized Flathead screwdriver.

##### 3.1.1 POWER RECONFIGURATION STEPS FOR 115V AC (-5V DC) OR -48V DC OPERATION

The DT-6061 is powered through either of two interfaces: -48V DC or -5V DC input. The interface used is determined by an option set at the factory that can be changed by the customer. Typically, the -5V DC interface is used when a customer uses an 115/240V AC to -5V DC power supply. The reseller of this product has tried to insure that the DT-6061 has been optioned at the factory for the power interface you need for your configuration.



After attaching power to the DT-6061, and the power light does not illuminate, to reconfigure the power input option, perform the following steps using the figure above as reference:

- Disconnect any power connectors to the Unit.
- Unscrew and remove the Unit cover exposing the top portion of the system board.
- With the face of the DT-6061 facing front, locate the jumper connector (*JP2*). Move, if needed, the jumper to the **5V** setting for the **AC to DC** converter power supply input or to the **48V** setting for direct **48V DC** power input. (*Refer to the figure.*)
- Replace the Unit cover and its associated screws, and reconnect the power.
- Verify the power light is lit.

##### 3.1.2 MOUNTING AND WIRING

The following items are needed when DT-6061 units are being installed

- DT-6061 unit(s)
- AC to DC power supply(s). -48 VDC is directly wired into the unit.



- ❑ Cables (see *Cabling section* for required Data and Console cable types and their configuration).
- ❑ A strain relief clamp for wire stabilization (*-48V DC operation ONLY*).
- ❑ An EIA standard 19-inch or 23-inch equipment rack with internal, vertical mounting rails. Hole spacing on the vertical, mounting rail must be 1.25 inches. Use the dimension specifications in Appendix E, section 15 to calculate how high the rack needs to be to support a specified number of DT-6061 units (*rack-mount configuration ONLY*).
- ❑ A pair of mounting brackets (*19" or 23"*) for each DT-6061 (*rack-mount only*).

### **3.1.2.1 STAND-ALONE DT-6061 (AC TO -5V DC POWER INPUT)**

- 1) Attach the provided feet to the bottom of the unit
- 2) Place the DT-6061 in its desired location such as a shelf in a data equipment rack.
- 3) Attach data transport cable(s) (see *Cabling section*).
- 4) Attach console cable by plugging 1 end of the RJ45 terminated, twisted pair, data cable into the DT-6061 console interface and the other into the port of the asynchronous device that will be used to configure or manage the DT-6061 (see *Cabling section*).
- 5) Plug the power supply into a standard 115/240V AC outlet and the barrel connector stemming from the power supply, into the circular connector on the DT-6061 faceplate labeled 5V DC.

### **3.1.2.2 RACK-MOUNT DT-6061 (AC TO -5V DC POWER INPUT)**

- 1) Attach the mounting brackets (*19" or 23"*) to each side of the DT-6061.  
*Note: Brackets have a left or right designation. With the DT-6061 facing forward the left bracket is attached to the left side of the DT-6061. Left-sided brackets are distinguished by the two pre-tapped screw holes residing on the front facing portion of the bracket.*
- 2) Fasten the DT-6061 to the equipment rack using appropriate screws (*typically supplied by the equipment rack manufacturer*).
- 3) Attach data transport cable(s) (see *Cabling section*).
- 4) Attach console cable by plugging 1 end of the RJ45 terminated, twisted pair, data cable into the DT-6061 console interface and the other into the port of the asynchronous device that will be used to configure or manage the DT-6061 (see *Cabling section*).
- 5) Plug the power supply into a standard 115/240V AC outlet and the barrel connector stemming from the power supply, into the circular connector on the DT-6061 faceplate labeled 5V DC.

### **3.1.2.3 STAND-ALONE DT-6061 (48V DC POWER INPUT)**

- 1) Attach the provided feet to the bottom of the unit
- 2) Fasten the strain relief to the left or right side of the DT-6061 (*utilizing two of the rack-mount bracket holes*).
- 3) Place the DT-6061 in its desired location such as a shelf in a data equipment rack.
- 4) Attach data transport cables (see *Cabling section*).
- 5) Attach console cable by plugging 1 end of the RJ45 terminated, twisted pair, data cable into the DT-6061 console interface and the other into the port of the asynchronous device that will be used to configure or manage the DT-6061 (see *Cabling section*).



- 6) Run your 48V DC (return, minus and ground) wires from a central source through the strain relief clamp (*for DC wire stabilization*). On the DT-6061 faceplate, attach the return, minus and ground wires to the return, minus and ground connections respectively of the terminal block labeled 48V DC.

#### **3.1.2.4 RACK-MOUNT DT-6061 (48V DC POWER INPUT)**

- 1) Attach the mounting brackets (19" or 23") to each side of the DT-6061.  
*Note: Brackets have a left or right designation. With the DT-6061 facing forward the left bracket is attached to the left side of the DT-6061. Left-sided brackets are distinguished by the two pre-tapped screw holes residing on the forward facing portion of the bracket.*
- 2) Fasten the strain relief to the DT-6061 rack mount bracket.
- 3) Fasten the DT-6061 to the equipment rack using appropriate screws (*typically supplied by the equipment rack manufacturer*).
- 4) Attach data transport cable(s) (*see Cabling section*).
- 5) Attach console cable by plugging 1 end of the RJ45 terminated, twisted pair, data cable into the DT-6061 console interface and the other into the port of the asynchronous device that will be used to configure or manage the DT-6061 (*see Cabling section*).
- 6) Run your 48V DC (*return, minus and ground*) wires from a central source through the strain relief clamp (*for DC wire stabilization*). On the DT-6061 faceplate, attach the return, minus and ground wires to the return, minus and ground connections respectively of the terminal block labeled 48V DC.



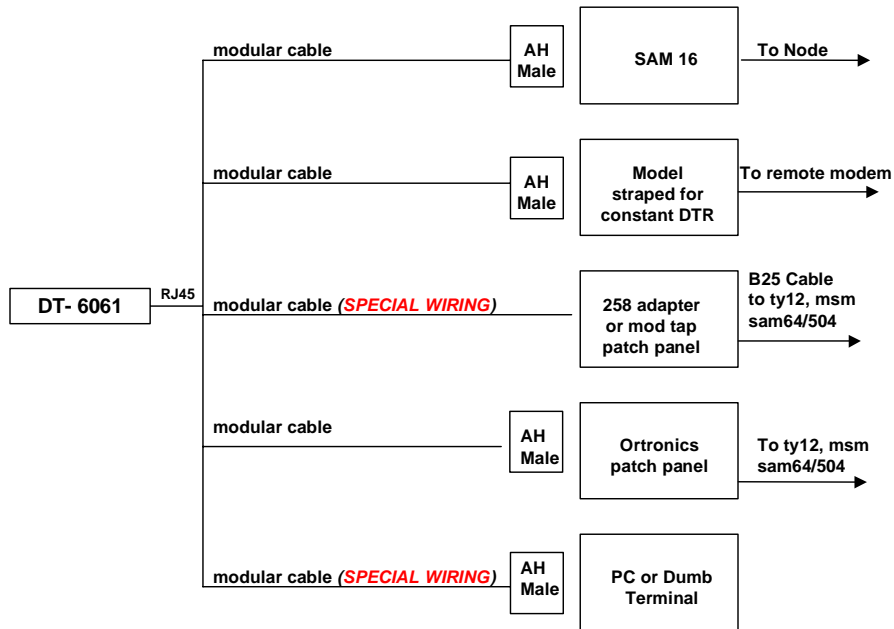
## 4 CABLING

### 4.1 SERIAL CONSOLE SETUP

The DT-6061 is managed through its serial console port by a terminal, PC, dial-up modem, or BNS asynchronous connection.

Specific instructions for configuration of SAM, TY12 and MSM asynchronous ports are available in the appropriate BNS module, reference guide. DT-6061 specific, configuration notes are described herein.

Console cables are available from your DT-6061 Reseller and may be required for console connection through BNS, TY12 and MSM modules, SAM64/504 Multiplexors and connection through an Ortronics distribution patch panel (see figure and table below).



*Important!* A modular cable with "SPECIAL WIRING" is different from a generic, 8 wire modular cable. The DT-6061 console will typically connect to a PC/dumb terminal or a BNS asynchronous device such as a TY12 or SAM. The diagram in Chapter 11 describes how to construct the with "SPECIAL WIRING". Otherwise, the cable can be ordered. Use the table below for part number information.

The following cables and adapters are available for console connections:

Cable or Adapter	Order Information (Lucent Technologies)	Order Information (CBM of America)
8 pin modular to 8 pin modular cable ( <i>standard</i> )	Comcode 408981653 (25')	P-10419-XX (XX ' length)
8 pin modular to 8 pin modular cable ( <i>special wiring</i> ) (25') - The wiring schematic is shown in Appendix B.	Comcode 408198133	DTCONCAB-25
Male 25 pin to 8 pin connector ( <i>AH Male</i> )	ED5P055-31 G (139)	ADPG139
258 Adapter	ED5P055-31 G (155)	ADPG155





**Important!** A modular cable with “SPECIAL WIRING” can be ordered using the table above or built using the wiring diagrams provided in this manual in section **Error! Reference source not found.** The pin-outs for the 9-pin connector is shown in section 0.

## 4.2 CONFIGURATION NOTES

- ❑ Configure SAM, TY12 and MSM console connections as 9600 bps with 8 bits and no parity, and use a DCE type cable.
- ❑ Configure SAM and MSM console connections as type “host” and as a “pap” (permanently active port).
- ❑ Configure TY12 console connections as type “console”.



## 5 QUICK START CONFIGURATION GUIDE

There are two levels of configuring a DT-6061: platform and application. This user manual covers only the platform configuration and registration, and the installation and registration of the application software. The commands and IP addressing associated with configuring an application in an instance are described in the corresponding User Manual for that application. Those manuals are available on the Datatek web site.

Below is a sample configuration sequence for a DT-6061 platform. All of these commands must be entered via the serial console connected to the CON port of the DT-6061..

### 5.1 PLATFORM CONFIGURATION

The following is the command sequence for the initial configuration of the DT-6061.

```
<DT-6061> login passwd=initial ↵   enter "logged: in mode
User is logged in
<DT-6061> date 11:05:00 09/22/2003 ↵enter date & time (optional)
Fri Sep 22 11:05:00 EDT 2000
<DT-6061> ipaddr 135.17.59.240 ↵   enter IP address of DT-6061
You must reboot for this change to take effect
<DT-6061> submask 255.255.255.0 ↵   enter Subnet Mask of DT-6061
You must reboot for this change to take effect
<DT-6061> gateway 135.17.59.1 ↵   enter Gateway Address of DT-6061 if necessary
You must reboot for this change to take effect
<DT-6061> reboot ↵
```

### 5.2 PLATFORM SOFTWARE REGISTRATION

The software for this product must be registered in order to run applications. Please call Datatek for a registration key. The key is based on the **product code** (DT-6061), the two-part software release number (**release.version**)<sup>1</sup> of the platform software, and the hardware **serial number** and **MAC address** of the DT-6061.

```
<DT-6061> register ↵
Product_Code=DT-6061
MAC=0.96.29.99.99.99
HW_SERNUM=99.99.99.99.99.99
Build_Number=16.0
Build_Date="Mon Feb  2 14:21:02 EST 2004"
```

```
Enter key: rxyuvw8s
Registration successful
```

### 5.3 APPLICATION INSTALLATION

This action requires that a host, acting as an ftp server, is present on a network that is accessible by the DT-6061 install command.

```
<DT-6061> install ↵
application or package name ? NAME=tn3270 ↵
name of ftp download server ? SRV=<server name> ↵
login on download server ? ID=<user's login> ↵
```

<sup>1</sup> Also called the **Build Number**.



```
password ? PASS=<user's passwd> ↵
name of directory on server ? LOC=(/dt6061/apps) <location of
application> ↵
Downloading "tn3270" ...
Downloaded:
-rwxrwxrwx 79701 Jun  5 05:21 tn3270.dt6
Testing ...
Sum of "tn3270": 34502, package type: DT-6061C
Installing ... .....
Install of "tn3270" successful
```

## 5.4 APPLICATION SOFTWARE REGISTRATION

Each application that is installed on a DT-6061 must be registered in order for it to run. Please call Datatek for a registration key. The key is based on the **application name**, the two-part software release number (**release.version**)<sup>2</sup> of the application software, and the hardware **serial number** and **MAC address** of the DT-6061. This information is readily available from the **regapp** command:

```
<DT-6061> regapp ↵
Unregistered applications:
  tn3270: Build 2.0 "Mon Jan 19 21:53:05 EST 2001"
  MAC Address = 0.96.29.111.14.29
  Serial Number = 17.61.14.111.14.29
application registration key ? KEY=larz7w8s
Registration successful for 'tn3270'
```

## 5.5 ASSIGNMENT OF AN APPLICATION TO AN INSTANCE

```
<DT-6061> app 1 type=tn3270 ↵      configuring instance #1 for installed application
Stopping 1 none(55)
PID 1033 running tn3270 1
```

---

<sup>2</sup> Also called the **Build** or **Build Number**.



## 6 PLATFORM COMMANDS

The DT-6061 software is composed of two components. One component, called the **Platform**, exists to support all applications. The second component is comprised of the individual **application(s)**.

The **Platform** provides Operating System functions, selected interfaces, protocol stacks, SNMP functions, and system OA&M while each **application** uses the services of the resident **Platform**.

Most commands can not be entered without first *logging in* with a user-settable password. The only commands allowed without first logging in are:

- help
- login
- rstpass

### 6.1 INPUT CONVENTIONS

All parameters may be given on the command line. Parameters of the form **name=<value>** may be given in any order.

For several complex commands, listed below, missing parameters, or corrections of errors in given parameters, of the form **name=<value>** are collected by prompting the console user. The user responds to a prompt for the **name** by typing the required **<value>** followed by *newline*. Defaults are supplied in some cases, so the user need only enter *newline*.

- ❑ The legal characters for command input are the upper and lower case letters, numbers, spaces, tabs, and the following special characters: **!#%+, - . / : = \_**.
- ❑ Commands may be entered in upper or lower case.
- ❑ Parameters of the form **name=value** may use upper or lower case for **name**.
- ❑ Default values, if any, are shown in parenthesis as part of the prompt.
- ❑ Case is preserved for values.
- ❑ Values may not contain spaces, tabs, or the = character.
- ❑ When a password is being requested by a prompt, input is not echoed. The legal characters for prompted passwords are the upper and lower case letters, numbers, and the following special characters: **!# \$%&' \*+, - . / : ; < > ? \_ { | } ~**.
- ❑ The **banner** and some of the parameters of the **snmp** command entered in response to prompts can use any special characters.
- ❑ Backspace erases one character and @ deletes the current line of input. Most commands are killed by the **del** key.
- ❑ An optional **timeout** can be configured for the console; if a command is not completely entered within the configured timeout period (in the range of 30 to 1000 seconds), the console is automatically logged off.

### 6.2 LOGIN

**Syntax:** login **PASSWD=<password>**

The **login** command is used to allow access to the other commands.



The **PASSWD** parameter is not echo suppressed. However, if the **PASSWD** parameter is not provided in the command line, the console prompts for a password; the response *is* echo-suppressed in this case.

If the password is valid, the user is placed in the *logged in* mode. Once the console user is logged *in*, the balance of the commands are accessible. The **login** command is not accessible if the user is already logged in.

## 6.3 LOGOUT

**Syntax:** `logout`

The **logout** command is only allowed if the console user is logged *in*. It uses no arguments. It will set the console to the logged *out* mode. The console may also be logged out by typing **exit** or **ctrl-D**.

## 6.4 DATE AND TIMEZONE

**Syntax:** `date [hh:mm:ss] [mm/dd/yyyy]`  
`timezone NAME=<timezone> START=<start time> STOP=<stop time>`  
`timezone help`

The **date** command sets the system date and time. Without arguments the date command displays the current date & time settings. Since the DT-6061 does not have a battery backed-up timing device, it may lose time across system reboots.

The **timezone** command configures the parameters for the user's time zone, for proper display of the date and time. It is intended that time be kept internally according to "Universal Time" (formerly known as Greenwich Mean Time). The time zone should be administered so that the **date** command and all reports that might print dates (e.g. timestamps on files) can display local time while the system uses Universal Time. The **help** option of **timezone** prints a description of how to set the timezone (similar to the following).

The **NAME** parameter takes the form *LBLhLBLh* (for example NAME=PST8PDT7), where the first *LBL* is the 3-letter label that designates "standard time" in that time zone, and the second *LBL* designates "daylight savings", if any, in that time zone. The first and second "*h*" values designate the number of hours west of UTC for standard and daylight savings time, respectively. These numbers may be expressed in one of three forms:

*n*        '*n*' hours west of UTC (-*n* if east)  
*h*:*m*    '*h*' hours and '*m*' minutes west of UTC (-*h*:*m*)  
*n.m*    '*n.m*' hours (i.e. decimal fraction) west of UTC

The given example, NAME=PST8PDT7, would be used for Pacific Standard Time with daylight savings. When daylight savings is NOT to be used, just enter the same values for the first and second "*LBL*" and "*h*", e.g. NAME=PST8PST8.

The **START** and **STOP** values give the date and time daylight savings starts and stops, according to the notation: *wDayMonTime*, where:

*Day*    the day of the week ('Sun', 'Wed', etc.)  
*Mon*    the month  
*w*        which such 'Day' of the month (e.g. '1' for the first, etc., and '5' denotes "last")  
*Time*    *h*:*m* is the time to switch to the new mode



The rule for most U.S. time zones (and the *default* for START and STOP) is:

```
START=1SunApr2:00 STOP=5SunOct2:00
```

which declares that daylight savings starts on the first Sunday of April at 2 a.m. and ends on the last Sunday of October at 2 a.m.

## 6.5 CHANGE PASSWORD

**Syntax:** `chgpas OLD=<old> NEW=<new> CONFIRM=<new>`

The **chgpas** command is used to change a user password on the system console. The command is only allowed if the user is logged *in*.

All three parameters can be given on the same line as the command. None of those entries are echo-suppressed. However, if parameters are omitted from the command line, the console will prompt for them, and the responses will be echo-suppressed. (Previous versions of the DT-6061 used **PASSWD** and **NEWPASS** instead of **OLD** and **NEW**; the old variations are still supported.)

If the current password is valid and the two entries for the new password match, the password is changed to the new value.

## 6.6 RESET PASSWORD

**Syntax:** `rstpass`

The DT-6061 has a user changeable password that is used to gain access to the console. If the password should become misplaced, console access would not be available for configuration and administration. The user may recover from this situation using the **rstpass** command. This command is always available regardless of whether the console is logged in or not. The **rstpass** command displays unique identification about this particular device along with a specially constructed access code and then prompts for the key. The user may then contact the manufacturer with the device information and access code to obtain the **software key**.

The **software key** is an eight character alphanumeric that is valid one time only for this particular DT-6061. If a valid key is entered the user password is reset to the original value of **initial**. If an invalid key is entered the message "**key is not valid for this system**" is generated.

## 6.7 SYSTEM PARAMETER CONFIGURATION

**Syntax:** `ipaddr <d.d.d.d> (ip address)`  
`submask <d.d.d.d> (subnet mask)`  
`gateway <d.d.d.d> (gateway router address)`

The above commands configure, or verify, the addressing and routing parameters of the DT-6061. When one of these commands is invoked without its argument, it will display the current configuration of the unit. When invoked with its argument, the current configuration is changed accordingly.

The **gateway** must belong to the same subnetwork as the module. That is, the part of the **gateway** address corresponding to the non-zero part of the **submask** must match the corresponding part of the **ipaddr**. If it is necessary to reconfigure a DT-6061 to a new subnetwork, first set the **gateway** to **none**, then change the **ipaddr**, then set the new **gateway**.

Changes to these system parameters do not become effective until after the next **reboot**.



## 6.8 SET CONSOLE TIMEOUT

**Syntax:** `timeout [ <number of seconds> | off ]`

The **timeout** command is only allowed when the unit is logged in.

The DT-6061 serial console uses a three-wire interface (RD, TD, GND), and the lead state of other signals is not relevant. This would imply that the only way to change the state of the console is to explicitly log in or log out, or reboot which forces the console to be logged out.

For users who wish the console to automatically log off after a period of inactivity, there is a console timer. The console timer defaults to the disabled condition, and may be activated by the **timeout** command.

The **<number of seconds>** value must be between 30 and 1000, inclusive. To check what the present value is, use the **timeout** command without any arguments, or see the timeout value in **vfy mod**.

When the DT-6061 determines that the period of inactivity of the specified time has elapsed, it automatically forces the console to log off. An **INFO**-level alarm is issued at that time.

## 6.9 SYSTEM LABEL CONFIGURATION

**Syntax:** `label [<word> (no spaces) | none]`

This command enters a *label* that subsequently appears as part of the system console prompt and is used as a key for backup and retrieve directories on the backup server. The word "**none**" deletes the label. (see *BACKUP/RETRIEVE*)

## 6.10 CONFIGURE THE BANNER

**Syntax:** `banner [edit]`

The **banner** command is only allowed if the unit is logged in.

Using the **banner** command, a banner that consists of up to 9 lines can be entered. The banner is displayed when the **login** command is executed. When the login command is entered with the password on the same line as the command, the banner is displayed after the login. If the **login** command is used in prompted mode, the banner appears before the password prompt.

When the **banner** command is executed with no parameters, the entire banner is displayed, and then the DT-6061 administrator may re-enter an entirely new banner line-by-line or use the 'DEL' key to exit without altering the existing banner. The entire banner can also be displayed by typing **vfy banner**.

To change one or more lines of text in the banner, the administrator types **banner edit**. The entire banner is then displayed with line numbers. The user is prompted for which line to change. When the user enters a line number, the system displays the corresponding banner line and prompts for a replacement line. The user must then type in the entire text for that line followed by a newline (enter). The user is again prompted for a line number. When finished, the user types a newline in response to the line number prompt.

The character set allowed for the banner is not restricted by the input rules of section 6.1. Any characters may be entered for the banner. The @, \, and *backspace* characters may be entered only by preceding them with \.

## 6.11 REBOOTING THE DT-6061

**Syntax:** `reboot`



After the DT-6061 is properly configured with system parameters, it must be restarted in order for the IP configuration to take effect. This is performed with the **reboot** command from the system console. The **reboot** command has no arguments. It will restart the system so that any new global parameters may take effect.

## 6.12 SYSTEM CONSOLE USER HELP

**Syntax:** `help [ ? [command] ]`

The available commands on the DT-6061 system console, along with a short narrative, is available via the **help** or **?** command. Command-specific help is available as well. For example, help "L" shows help for all commands that begin with "L" or help "login" shows help for this specific command.

## 6.13 CONFIGURE HOST NAMES

**Syntax:** `hosts add|del IP=d.d.d.d name=<name> ... name=<name>`

The **hosts** command is only allowed when the unit is logged in.

The **hosts** command is used to configure the hosts translation table.

The DT-6061 can maintain a set of domain-style mnemonic names for originating calls to the IP network, analogous to the /etc/hosts file on both UNIX and Microsoft Windows platforms. This allows the DT-6061 to perform a translation between a user-provided domain-style name and its associated IP address during call setup without using a DNS server. The use of a mnemonic name is optional; the DT-6061 will always accept an IP address in its base form. If the name is not present in the **hosts** table, then the resolution of the name to an IP address is attempted via the DNS servers specified in the **dns** command below.

The **name** parameter value is a mnemonic of 24 characters or less in length, using upper and lower case letters, digits, and the '-' (hyphen) and '.' (period) characters. Upper and lower letters are treated the same. That is, the name value is case insensitive. More than one name can have the same resultant IP address. When using the **del** option, the reserved value **all** can be used as the name value to delete all names associated with a particular IP address. If all names are deleted for an IP address, then the IP address is removed from the table as well.

## 6.14 CONFIGURE DNS (DOMAIN NAME SERVER)

**Syntax:** `dns [ name<n>=<name> ] [ ipaddr<n>=d.d.d.d ] (n=1,2,3)`

The **dns** command is only allowed when the unit is logged in.

The **name1**, **name2**, and **name3** parameters are domain names. These domain names are appended to a domain-style name that is not fully specified for DNS purposes. For example, a name "bender.ho.lucent.com" is fully specified, so nothing is appended by the DT-6061. A name such as "bender" would need to have a domain appended before the DNS server could resolve it. The DT-6061 will append the specified domain names in the order of **name1** through **name3**, and send the resulting strings to the DNS server in succession until the latter is able to perform a resolution. The name can consist of upper and lower case letters, hyphen, digits, and the '.' (period) character. The names are case insensitive and can be any length.

Each **ipaddr<n>** field is the IP address of a Domain Name Server to be used to resolve domain-style addresses not defined in the **hosts** table. The DNS IP addresses are used in the order specified. If only one address is to be defined, it should be **ipaddr1**.

Each time the **dns** command is run, the user is prompted for all values except for those specified on the command line. The user is prompted in the following order: **name1**, **name2**, **name3**,





**ipaddr1, ipaddr2, ipaddr3.** In order to delete a value, reply to a prompt with the word “**none**” or on the command line, type the parameter name with no value following; for example,

```
dns ipaddr1= or dns name2=
```

## 6.15 OPERATIONAL APPLICATIONS CONFIGURATION

**Syntax:** `app <instance> [cnt=<number>] [type=<type>|none]`

As noted earlier, the DT-6061 simultaneously supports many *instances* of different applications. The number and type of application must be configured as a system parameter. This is done with the **app** command. The **app** command will allow the installation of any supported and properly installed application into an *instance*. There are **N** such instance slots available (*currently 30*). Any combination is supported.

The **<instance>** is the location where the selected application is to be installed. There exist **N** such instances on the DT-6061. If there was a prior application in the location specified, it is replaced by the new application.

The **cnt=<number>** is an optional parameter that allows multiple consecutive instance locations to be configured identically. When this parameter is not used, the number of instances affected by the command is one. (Previous versions of the DT-6061 used **num=** for this purpose. The older syntax is still supported.)

The **type=<type>** is a required parameter that defines the application to be installed in the instance specified. If the value is **NONE**, then any application in the instance specified is deleted (including its configuration). The **type** must be the name of an application previously installed on the DT-6061 by the **install** command.

## 6.16 RESTART AN INSTANCE

**Syntax:** `restart <instance> ...`

The **restart** command is used to restart an application instance. This action will stop any data transport for that particular instance and disconnect any TCP connections, including the application's console. The **<instance>** parameter is used to specify the application instance number to be re-started. Multiple **<instance>** numbers may be given in one command to restart multiple instances.

## 6.17 REMOVE OR RESTORE AN INSTANCE

**Syntax:** `remove <instance> [ cnt=<num> ]`  
`restore <instance> [ cnt=<num> ]`

The **remove** (or **rm**) and **restore** (or **rs**) commands may be used to suspend and resume the operations of one or more instances without changing or losing the configurations associated with those instances.

The **<instance>** parameter is the application instance to be removed from or restored to service. Multiple instance numbers may be given in one command. The optional **cnt=<num>** parameter specifies some consecutive number of instances to be removed or restored.

While an instance is out of service, it maintains no connections with any other network equipment. Its console is also out of service, and thus no configuration changes may be made in that instance. The **vfy app** and **vfy cfg** commands show which instances are out of service.



## 6.18 RESET AN APPLICATION PASSWORD

**Syntax:** `rstapp <instance>`

Each application instance is allowed to have its own user changeable password for configuration purposes. In the event that the password is lost, it may be necessary to reset this password back to the initial value. Normally, this could be done by deleting the application instance and re-installing one of the same type. However, that approach would also delete all configuration of that application instance, and would be disruptive to the operation of the components served by the application instance. The **rstapp** command resets the application instance password by deleting its password file. This resets the password back to the value *initial*.

The **<instance>** is the application instance for which the password is to be reset. The instance must be removed and restored to service for the password change to take effect.

## 6.19 MANAGE LIST OF COMPANION DT PRODUCTS

**Syntax:** `dtproduct: add|del <ipaddr> [<ipaddr> ...]  
vfy dtproduct`

In this version of the platform software, coupled with newer software versions of DT-6061 applications and other Datatek products, TCP connection breakage is detected automatically. The **dtproduct** command is no longer necessary when the user's network is running newer software versions.

The **dtproduct add** command adds the given IP addresses to a list of network elements to be treated specially whenever the DT-6061 reboots (or, in the case of a redundant DT-6061, when switchover between *active* and *standby* occurs). The special treatment accelerates the reestablishment of TCP connections between applications on the DT-6061 and those network elements. Only Datatek-manufactured IP-based network elements, such as the DT-4000 and the UMI, are able to respond to this acceleration treatment. IP addresses are deleted from this list by **dtproduct del**.

Under various conditions of equipment and network outage, it can sometimes take time (minutes) for IP connections to close and reconnect. This special acceleration helps reduce downtime for those connections between a rebooting DT-6061 and other Datatek products. To enable the DT-6061 to use this feature, the user uses the *optional* **dtproduct** command to build a list of IP addresses for which this feature is to be used. Properly configured, this feature helps reestablish the connections between a DT-6061 and other Datatek network elements very quickly after a DT-6061 is rebooted. Non-Datatek equipment will not respond to this treatment.

The **dtproduct** list may be viewed with the **vfy dtproduct** command.

*Note: This is an optional configuration to improve performance and reliability after network equipment outages, and is not required for correct functionality.*

**Here are some guidelines to decide when to use the dtproduct command with older software versions of these products:**

- ❑ If any application on the DT-6061 has been configured to originate calls to a DT-4000 or UMI with IP address 'A', then it is beneficial to add 'A' to the **dtproduct** configuration.
- ❑ If a DT-4000 or UMI with IP address 'B' has been configured to originate calls to this DT-6061, then it is beneficial to add 'B' to the **dtproduct** configuration for this DT-6061.
- ❑ If a DT-4000 or UMI at address 'C' has been permanently removed from the network or no longer has connections configured with this DT-6061, it is beneficial to delete 'C' from the **dtproduct** configuration of this DT-6061.



- ❑ The absence of one or more DT-4000's from the network when they occur in the **dtproduct** list on a DT-6061 can cause up to a 10-second (total) delay when rebooting that DT-6061.
- ❑ There is very little overhead in adding 'D' to a **dtproduct** configuration of a DT-6061, as long as the DT-4000 associated with 'D' is usually up and running when this DT-6061 is rebooted; otherwise the 10 second boot delay described above may apply. However, a list larger than 40 or 50 should be carefully managed.

## 6.20 CONFIGURE SNMP

**Syntax:** `snmp get=<communities> set=<communities>  
trap=<community> dest=<IP address> cug=[+|]<cuglist>  
(prompted: sysName=<esc> sysLoc=<esc> sysContact=<esc>)`

The **snmp** command is only allowed when the unit is logged in.

The **snmp** command configures or verifies the parameters needed to permit the SNMP agent to communicate with a manager using closed user groups and user-specified communities for **gets** and **sets**. The DT-6061 will not respond to other community names. The **<communities>** parameter accepts a comma-separated list of community names. The communities configured for **set** also work for **get**, so they need not be repeated in the **get** parameter. The closed user group option (**cug**) defines which IP addresses are allowed to do **gets** and **sets**. Values are saved in non-volatile memory. There is only one **trap=<community>**. The **dest=<IP address>** parameter configures the IP address of the target trap manager. The **snmp vfy** command (same as **vfy snmp**) displays the current settings of SNMP configuration.

The **snmp** command prompts for all parameters not specified on the command line. Simply type **snmp** in order to add or change any parameters. If the present value that is printed is to be retained, hit newline in response to the prompt. In order to add to an existing list for the **set** and **get** communities, type in the old values as well as the additions for that parameter when prompted. To delete all values for a keyword, type the word "none" as the value. Special characters and escape sequences are accepted for community strings, **sysName**, **sysLoc**, and **sysContact** parameters, but must be entered at the prompt for that parameter, not on the command line.

When the SNMP configuration is changed, the agent is restarted. Whenever the DT-6061 reboots or the agent is restarted, a warm-start trap is sent to the configured trap manager. The only other trap sent by the DT-6061 platform agent is "authentication fail" when a manager attempts to use a community name not on the list. This trap is only sent if a manager has enabled it to be sent.

The **snmp sysName**, **sysLoc**, and **sysContact** are changeable via an SNMP manager as well as via the console.

The enterprise object ID of the DT-6061 platform is **1.3.6.1.4.1.3791.3.7**.

## 6.21 CLOSED USER GROUPS

**Syntax:** `cug <cugnum>[ipaddr=<d.d.d.d>][submask=<IP submask>]`

The **cug** command is only allowed when the unit is logged in.

The **<cugnum>** parameter is the closed user group identifier used to assign the CUG to the SNMP interface (with the **SNMP** command). If SNMP is configured with one or more CUGs, only callers belonging to those CUGs are allowed to issue GETs and SETs. If no CUG is assigned to SNMP, any caller is allowed to do SETs and GETs. (SNMP also requires adherence to the community strings configured for SET and GET.)

Each CUG is specified by a single IP address and subnet-mask pair. The **ipaddr** parameter is an IP address or the base address of an IP subnetwork that identifies members of the group. The



caller's IP address *AND'ed* with the **submask** must agree with a CUG's **ipaddr** value *AND'ed* with the same **submask** value for the caller to belong to the CUG. Depending on the **submask** value, this allows an individual (submask=255.255.255.255), intermediate, or network-wide level of authorization.

Setting the **ipaddr** value to 0.0.0.0 deletes any prior configuration for the **<cugnum>**. A **<cugnum>** may not be deleted if it is currently assigned to SNMP.

A list of all configured CUGs is reported via the **vfy cug** command. The list of closed user groups associated with SNMP is displayed as part of the **vfy snmp** command.

## 6.22 VERIFY CONFIGURATION SETTINGS - VFY

The **vfy** command is used to report configuration settings for the following categories using the indicated syntax:

- **vfy app** - application installation and run-time status
- **vfy banner**
- **vfy cfg ...** - application instance selection
- **vfy cug** - closed user group definitions
- **vfy dtproduct**
- **vfy mod** - module configuration and status
- **vfy dns** - domain name server configuration
- **vfy hosts** - host-IP translation table
- **vfy snmp** - snmp settings

The verify reports for **banner**, **cug**, **dtproduct**, **dns**, **hosts**, and **snmp** are simple representations of the configuration parameters described in each corresponding command. The more complex verify reports are explained in this section.

### 6.22.1 VERIFY CONFIGURED APPLICATION INSTANCES

**Syntax:** **vfy cfg [<instance> [cnt=<num>]]**

The **vfy cfg** (or **vfy cfg**) command displays each configured instance, and what application type is configured on that application instance.

The **<instance>** parameter is an optional argument to specify the beginning instance number to begin the display. If not specified, it defaults to the number one. The report displays the instance number and application installed on that number. If the **cnt=<num>** parameter has also been specified, that number of *configured* instances will be displayed starting from the given instance number. (Previous versions of the DT-6061 used **num=** for this purpose. The older syntax is still supported.) With no parameters, all configured instances are displayed.

### 6.22.2 VERIFY RUNNING APPLICATIONS

**Syntax:** **vfy app**

The **vfy app** (or **vfy app**) command shows the list of installed applications, along with their build (revision) numbers and build dates. Unregistered applications are displayed with the notation "not registered." The display continues with a list of applications currently running in each application slot (*instance*), including "none", along with an internal identifier. Instances that are removed from services will show an "oos" status. Instances configured for an unregistered application will show



an "unreg" status. For some applications, there may occasionally appear a short temporary status report generated by some instances of the application.

### 6.22.3 VERIFY MODULE

**Syntax:** `vfy mod`

The `vfy mod` (or `vfymod`) command shows the DT-6061 module parameters and properties: Registration status, system label, IP address, subnet mask, gateway IP address, the redundancy status and configuration, if any, host name, MAC address, serial number, the manufactured "right to use" value, the platform software revision number and date, and the elapsed time since the system was booted. If platform installation is incomplete, an additional line of information shows the next action required. If the platform software is not registered, an additional line says so. If pending system reconfiguration requires a reboot, an additional line says so.

## 6.23 DISPLAY CONNECTIONS

**Syntax:** `dc [ -a ]`  
`dc app [ -f ] [ -a ]`

The DT-6061 may have thousands of TCP connections to other elements. A complete list of these connections is reported by the `dc` (or `dconn`) command. The optional `-a` causes the report to include the listening ports awaiting connection requests. The following is a `dc` report example:

```
< DT-6061> dc
```

Active Internet connections

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	(state)
tcp	0	0	192.168.8.250.10002	135.17.59.198.34052	ESTABLISHED
tcp	0	0	192.168.8.250.10010	135.17.59.198.34054	ESTABLISHED
tcp	0	0	192.168.8.250.1340	192.168.8.15.31000	SYN_SENT
udp	0	0	192.168.8.20.1030	*.*	
udp	0	0	192.168.8.20.9998	*.*	

In the above report, "Recv-Q" and "Send-Q" indicate the number of bytes currently buffered for receive processing, or queued for transmission, respectively. Note that no foreign address is shown in the case of UDP protocol, because UDP is a connectionless protocol. The following are the possible TCP connection states:

BOUND	Bound to a local address, ready to connect or listen.
CLOSED	Closed. The socket is not being used.
CLOSING	Closed, then remote shutdown; awaiting acknowledgment.
CLOSE_WAIT	Remote shutdown; waiting for the socket to close.
ESTABLISHED	Connection has been established.
FIN_WAIT_1	Socket closed; shutting down connection.
FIN_WAIT_2	Socket closed; waiting for shutdown from remote.
IDLE	Idle, opened but not bound.
LAST_ACK	Remote shutdown, then closed; awaiting acknowledgment.



LISTEN	Listening for incoming connections.
SYN_RECEIVED	Initial synchronization of the connection underway.
SYN_SENT	Actively trying to establish connection.
TIME_WAIT	Wait after close for remote shutdown retransmission.

A variation of the **dc** command:

**dc app [-f] [-a]**

reports only connections that are associated with *instances*, and shows the *instance* ownership of each connection. The list is sorted by local port number. The optional **-f** flag sorts the list by foreign IP address and port number. The following is a **dc app** report example:

```
< DT-6061> dc app
Active Internet Connections by Applications (sorted by local address):
  Type Inst FD  Local Addr                Foreign Addr
b2aprt 10  .5  192.168.8.250.1340        192.168.8.15.31000
tn3270  2  .5  192.168.8.250.10002       135.17.59.198.34052
b2aprt 10  .4  192.168.8.250.10010       135.17.59.198.34054
```

Note that in the above report, "FD" (file descriptor) is only for use during software development or troubleshooting.

The connections may also be viewed from an SNMP manager.

## 6.24 DISPLAY MEASUREMENTS

**Syntax: dm [-d] ip | tcp | udp | icmp | con**

The DT-6061 platform maintains measurements of network activity as an aggregate of all the application instances. The measurements are grouped into five categories. The measurements for any one of the categories may be reported by the **dm** command, giving one of the categories as a parameter. The categories IP, TCP, UDP, and ICMP report measurements for that protocol. The CON report shows the current number of connections. The optional **-d** reports the differences between the current measurements and the previous results from **dm**. There is no way to clear the measurements.

These measurements may also be viewed from an SNMP manager. Appendix D, section 14 shows sample reports along with information about the individual measurements for the four protocol categories.



## 6.25 BACKUP AND RETRIEVE APPLICATION CONFIGURATION DATA

The backup command uses **ftp** to send a package containing the entire application configuration to the user's backup server. The retrieve command can retrieve all of the configuration or just the portion associated with some instance(s).

```
BACK SRV=<IP Address>
  [ID=<Login ID if not anonymous>]
  [PASS=<Password if not anonymous>]
  [LOC=<Path on FTP Server if not /dt6061/backups/<label>>]
  [FILE=<backup filename>]

RETR all SRV=<IP Address>
  [ID=<Login ID if not anonymous>]
  [PASS=<Password if not anonymous>]
  [LOC=<Path on FTP Server if not /dt6061/backups/<label>>]
  [FILE=<backup filename>]

RETR <instance#> [ <instance#> ... ] SRV=<IP Address>
  [ID=<Login ID if not anonymous>]
  [PASS=<Password if not anonymous>]
  [LOC=<Path on FTP Server if not /dt6061/backups/<label>>]
  [FILE=<backup filename>]
```

The backup package saves platform and application configuration. Platform configuration comprises the list associating instance numbers with application types, the application keys, the *dns* and *hosts* configuration, the *snmp* configuration, the console *timeout*, the *banner*, and the *dtproduct* list.

The IP address of the server may be given as a domain-style address if the DT-6061 has been configured with a working DNS or HOSTS configuration. The default **<label>** above is either the *IP* address of the DT-6061 or the configured *label*. The user may choose a unique name for FILE (the default name is *bkup*).

The *back* and *retr* commands remember the previous SRV, LOC, and FILE values and provide them next time as defaults for the SRV, LOC, and FILE prompts.





## 7 DIAGNOSTICS

### 7.1 PING

**Syntax:** ping <IP Address>

The **ping** command verifies connections to remote computer or computers, by sending 3 ICMP echo packets, one second apart, to the remote computer and listening for echo reply packets. The IP address can use a domain-style address, providing the **dns** and/or **hosts** configuration has been properly entered.

### 7.2 TRTE ( OR TRACERT)

**Syntax:** trte <IP Address>

The **trte** command determines the route taken to a destination by sending Internet Control Message Protocol (ICMP) echo packets with varying Time-To-Live (TTL) values to the destination. Each router along the path is required to decrement the TTL on a packet by at least 1 before forwarding it, so the TTL is effectively a hop count. When the TTL on a packet reaches 0, the router is supposed to send back an ICMP Time Exceeded message to the source system. Tracert determines the route by sending the first echo packet with a TTL of 1 and incrementing the TTL by 1 on each subsequent transmission until the target responds or the maximum TTL is reached. The route is determined by examining the ICMP Time Exceeded messages sent back by intermediate routers.

The IP address can use a domain-style address, providing the **dns** and/or **hosts** configuration has been properly entered.

### 7.3 DIAG COMMAND

**Syntax:** diag <subcommand> <parameters>

The **diag** command reports a variety of reports and statistics from the DT-6061 console and can aid in network monitoring and troubleshooting. **Diag** is provided at the <DT-6061> prompt, but is not currently displayed with the help command. Using the **diag** command with no recognized suboptions results in the message:

```
diag: Invalid command
```

Here are the various subcommands of **diag**:

#### 7.3.1 ADDRESS

**Syntax:** diag addr <IP Address> <port>

The **addr** subcommand reports the instance associated with the supplied IP address and port. If the <port> parameter is **0**, all instances with ports associated with the supplied IP address are reported.

#### 7.3.2 CPU

**Syntax:** diag cpu

The **cpu** subcommand reports *CPU* utilization by process. Processes may be associated with instance numbers as reported by the **vfyapp** command.

#### Sample Output

```
diag cpu
0: 398711    99%
1: 5014     1%
```





```

Interrupts stopped: 0 started: 0
  init   - k:      13 u:      0
  none   - k:       1 u:       1
  onsite  - k:       5 u:       4
  none%0  - k:       3 u:       2
  none%1  - k:       1 u:       1
  none%2  - k:       2 u:       1
  none%3  - k:       2 u:       0
  .
  .
  .
  tn3270  - k:     144 u:     59
  tn3270%0 - k:     146 u:     63
  cme     - k:       0 u:       0
  none%a  - k:       4 u:       1
  tn3270%1 - k:    115 u:     66
  tn3270%2 - k:    119 u:     35
  none%b  - k:       3 u:       1

```

**Notes:**

- First line (0:) shows total kernel plus idle time\*
- Second line (1:) shows total application time\*
- Subsequent lines show kernel (k:) and application (u:) time\* for each thread.
- Thread names contain an application type name or a system thread name. (There is no relation between thread names and particular instance numbers.)

*All times are in 100'ths of a second*

**7.3.3 DEV**

**Syntax:** `diag dev` (use lower case)

The **dev** subcommand reports CPU utilization and interrupt counts by category, including device drivers, system, applications, and scheduler (idle). The command prints utilization percentage and interrupt counts *since the last dev command*.

**Sample Output**

```

diag dev
*** IV ***
  con: 0          0% 89
  sch: 629       0% 121120
  sclk: 0        0% 113992
  eth0: 132     0% 8912
  SW+0: 127     0% 4682
  SW+1: 0       0% 5700
  SW+2: 0       0% 2280
  sys: 4886     1%
  app: 305      0%
  sched: 407077 99%

```

**Notes:**

Each line represents a gross category of CPU usage. The numbers in each line are:



- total CPU time used (in 100'ths of seconds) since boot (can wrap around)
- percentage of available CPU time used since the previous 'diag dev'
- the count of events since the last 'diag dev'.

*The most useful part of the report is the percentage (2nd number).*

Category key:

con - arriving system console characters (very small)  
sch & sclk - system scheduler clock ticks  
eth0 - low-level handling of transmitted or received packets  
SW+0,1,2 - handling TCP traffic  
sys - total time used by applications in common platform code  
app - total time in application code  
sched - idle time  
(*sys, app, and sched do not have event counts*)

The elapsed time since the previous 'diag dev' can be derived from the 'sclk' category report; its event count represents clock ticks, or 1/100ths of seconds.

### 7.3.4 PING

**Syntax:** `diag ping <IP Address>`

The **ping** subcommand is the same as the **ping** command.

### 7.3.5 TRACERT

**Syntax:** `diag tracert <IP Address>`

The **tracert** subcommand is the same as the **trte** command.



## 8 SNMP

The DT-6061 SNMP agent supports the ability to request the values of one or more Management Information Base (MIB) variables with the SNMP **Get** command.

### 8.1 DT-6061 SNMP MIB VARIABLE DATABASE

MIB Variable	Name	MIB
1.3.6.1.2.1.1.1.0	SysDescr	MIB-II
1.3.6.1.2.1.1.2.0	SysObjectID	MIB-II
1.3.6.1.2.1.1.3.0	SysUpTime	MIB-II
1.3.6.1.2.1.1.4.0	SysContact	MIB-II
1.3.6.1.2.1.1.5.0	SysName	MIB-II
1.3.6.1.2.1.1.6.0	SysLocation	MIB-II
1.3.6.1.2.1.1.7.0	SysServices	MIB-II
1.3.6.1.2.1.4.1.0	IpForwarding	MIB-II
1.3.6.1.2.1.4.2.0	IpDefaultTTL	MIB-II
1.3.6.1.2.1.4.3.0	IpInReceives	MIB-II
1.3.6.1.2.1.4.4.0	IpInHdrErrors	MIB-II
1.3.6.1.2.1.4.5.0	IpInAddrErrors	MIB-II
1.3.6.1.2.1.4.6.0	IpForwDatagrams	MIB-II
1.3.6.1.2.1.4.7.0	IpInUnknownProtos	MIB-II
1.3.6.1.2.1.4.8.0	IpInDiscards	MIB-II
1.3.6.1.2.1.4.9.0	IpInDelivers	MIB-II
1.3.6.1.2.1.4.10.0	IpOutRequests	MIB-II
1.3.6.1.2.1.4.11.0	IpOutDiscards	MIB-II
1.3.6.1.2.1.4.12.0	IpOutNoRoutes	MIB-II
1.3.6.1.2.1.4.13.0	IpReasmTimeout	MIB-II
1.3.6.1.2.1.4.14.0	IpReasmReqds	MIB-II
1.3.6.1.2.1.4.15.0	IpReasmOKs	MIB-II
1.3.6.1.2.1.4.16.0	IpReasmFails	MIB-II
1.3.6.1.2.1.4.17.0	IpFragOKs	MIB-II
1.3.6.1.2.1.4.18.0	IpFragFails	MIB-II
1.3.6.1.2.1.4.19.0	IpFragCreates	MIB-II
1.3.6.1.2.1.4.21.0	IpRoutingDiscards	MIB-II
1.3.6.1.2.1.5.1.0	IcmpInMsgs	MIB-II
1.3.6.1.2.1.5.2.0	IcmpInErrors	MIB-II
1.3.6.1.2.1.5.3.0	IcmpInDestUnreach	MIB-II

1.3.6.1.2.1.5.8.0	IcmpInEchos	MIB-II
1.3.6.1.2.1.5.9.0	IcmpInEchoReps	MIB-II
1.3.6.1.2.1.6.1.0	TcpRtoAlgorithm	MIB-II
1.3.6.1.2.1.6.2.0	TcpRtoMin	MIB-II
1.3.6.1.2.1.6.3.0	TcpRtoMax	MIB-II
1.3.6.1.2.1.6.4.0	TcpMaxConn	MIB-II
1.3.6.1.2.1.6.5.0	TcpActiveOpens	MIB-II
1.3.6.1.2.1.6.6.0	TcpPassiveOpens	MIB-II
1.3.6.1.2.1.6.7.0	TcpAttemptFails	MIB-II
1.3.6.1.2.1.6.8.0	TcpEstabResets	MIB-II
1.3.6.1.2.1.6.9.0	TcpCurrEstab	MIB-II
1.3.6.1.2.1.6.10.0	TcpInSegs	MIB-II
1.3.6.1.2.1.6.11.0	TcpOutSegs	MIB-II
1.3.6.1.2.1.6.12.0	TcpRetransSegs	MIB-II
1.3.6.1.2.1.6.13.X	TcpConnTable	MIB-II
1.3.6.1.2.1.6.14.0	TcpInErrs	MIB-II
1.3.6.1.2.1.6.15.0	TcpOutRsts	MIB-II
1.3.6.1.2.1.7.1.0	UdpInDatagrams	MIB-II
1.3.6.1.2.1.7.2.0	UdpNoPorts	MIB-II
1.3.6.1.2.1.7.3.0	UdpInErrors	MIB-II
1.3.6.1.2.1.7.4.0	UdpOutDatagrams	MIB-II
1.3.6.1.2.1.7.5.X	UdpEntry Table	MIB-II
1.3.6.1.2.1.11.1.0	SnmpInPkts	MIB-II
1.3.6.1.2.1.11.3.0	SnmpInBadVersions	MIB-II
1.3.6.1.2.1.11.4.0	SnmpInBadCommuni	MIB-II
1.3.6.1.2.1.11.5.0	SnmpInBadCommuni	MIB-II
1.3.6.1.2.1.11.6.0	SnmpInASNParseErr	MIB-II
1.3.6.1.2.1.11.30.0	SnmpEnableAuthenT	MIB-II
1.3.6.1.2.1.11.31.0	SnmpSilentDrops	MIB-II
1.3.6.1.2.1.11.32.0	SnmpProxyDrops	MIB-II

The SNMP object ID for the product code is the following for each of the DT-6XXX products:

1. DT-6061 - 1.3.6.1.4.1.3791.3.7
2. DT-6160 - 1.3.6.1.4.1.3791.3.13
3. DT-6260 - 1.3.6.1.4.1.3791.3.14



## 9 APPLICATION AND PLATFORM SOFTWARE INSTALLATION & UPGRADE

### 9.1 INSTALL AN APPLICATION

**Syntax:** `install NAME=<app-list> SRV=<host> ID=<host login>  
PASS=<password> LOC=<directory on host>`

An application may be initially installed, or upgraded, using the **install** command.

*Note: The application software package file must first be placed on an FTP server accessible to the DT-6061.*

The **install** command is invoked from the RS-232C system console and accepts:

- ❑ The **NAME** parameter is a comma-separated list of application names or platform update names.
- ❑ The **SRV** parameter is the IP address of the FTP server. It may be entered as a domain-style address if the DNS and/or HOSTS configurations have been established.
- ❑ The **ID** parameter is the user id to be used on that FTP server.
- ❑ The **PASS** parameter is the password associated with the user id on the FTP server
- ❑ The **LOC** parameter is the location (*directory name if not /dt6061/apps*) of the DT-6061 application files on the FTP server,

The **install** command will then operate as an FTP client to acquire the application files from the FTP server. Once the application files have been acquired, the **install** command will disconnect from the FTP server, and unpack the application into its proper locations on the DT-6061. In doing so, the DT-6061 will display the *sum* of each application or platform update, which may be compared with the sum documented on the Datatek download web page.

If parameters are omitted from the command line, **install** will prompt for them. When prompted for the password, input is not echoed. Note that it is necessary to specify **ID** and **PASS** only if the FTP server is not a standard anonymous FTP.

The *install* command remembers the last SRV and LOC and provides them as defaults for the SRV and LOC prompts.

Multiple applications may be downloaded from the same SRV and LOC in a single invocation of *install*. This is done by listing all the application names separated by commas, with NO spaces, as the value of NAME. E.g.: `install NAME=tn3270,x25pad` etc.

This release of the DT-6061 can install software packaged for the newer Embedded Network Applications Processor hardware, the DT-6160 and DT-6260, while also supporting software packaged for DT-6061 platform software version 15. All future applications will be packaged for use on the DT-6160, DT-6260, and DT-6061 release 16.0 or greater. New applications cannot be installed in the DT-6061 running version 15 or earlier.

### 9.2 APPLICATION SOFTWARE REGISTRATION

**Syntax:** `regapp [ <key> ]`

Each application that is installed or upgraded on a DT-6061 must be registered in order for it to run. Please call Datatek for a registration key. The key is based on the **application name**, the software version (the two-part **build number**, e.g. 9.1) of the application software, and the



hardware **serial number** and **MAC address** of the DT-6061. An unregistered application may be assigned to instances, but they will not start until the application is registered.

The **regapp** command accepts a key on the command line. If the key matches an installed application, that application becomes registered. If a key is not given on the command line, a list of unregistered applications is displayed followed by the module hardware information. Then the user is prompted for a key.

### 9.3 UNINSTALL AN APPLICATION

**Syntax:** `uninstall NAME=<app>`

The **uninstall** command is used to make room to install a new application type.

### 9.4 INSTALLING A PLATFORM UPGRADE

Platform upgrades use the **install** command using the same approach as application installation, with platform upgrades distinguished from application installation in the naming of the upgrade files. Platform upgrades are always named **basexxxx**. Before upgrading the platform, users should study the installation procedures in the release notes for the given upgrade, because the order of installation and the sequence of rebooting between installation steps is **very important**, and may be different from one release to the next.

### 9.5 PLATFORM SOFTWARE REGISTRATION

**Syntax:** `register [ <key> ]`

When a new release of the platform software is installed or when a new DT-6061 is purchased with software already installed, the software must be registered. Please call Datatek for a registration key. Until the platform software is registered, all DT-6061 console commands are still effective, but applications will not run.

The **register** command accepts the key on the command line. The key is based on the **product code** (DT-6061), the software version (the two-part **build number**, e.g. *16.0*) of the platform software, and the hardware **serial number** and **MAC address** of the DT-6061. If the key is not given on the command line, the information required for registration is displayed and the user is prompted to enter the key. If the key entered is correct, any registered applications that have been previously assigned to instances will start.



## 10 HIGH AVAILABILITY OPTION

The High Availability option couples two DT-6061 modules as an *active* and a *standby*. The *active* DT-6061 runs applications, while the *standby* "stands by" and monitors the *active* DT-6061. If the *active* DT-6061 fails or loses power, the *standby* will shortly take over the *active* role.

In *standby* mode, the DT-6061 has completed system boot, but has not started applications. A *standby* DT-6061 supports a subset of the commands in section 6, plus a small number of commands for support of high availability, documented later in this section. The prompt at the *standby* system console contains the word "standby".

Whenever a DT-6061 with high availability configuration boots, it first enters *standby* mode and begins to poll the other DT-6061 every few seconds across the IP network that joins them. There are three possible outcomes:

- The other DT-6061 doesn't respond (within a period of approximately 20 seconds), so this *standby* becomes *active*.
- The other DT-6061 is already *active*, so this *standby* updates its internal time-of-day from the *active* and continues to poll.
- The other DT-6061 is also in *standby*, so a brief negotiation follows, and one of them becomes *active*, while the other remains in *standby*.

The *standby* periodically polls the *active* and also (optionally) updates its version of system and application configuration from the *active*. If the *active* DT-6061 fails to respond to probes from the *standby* (for approximately 25 seconds), the *standby* takes over as *active* by starting up the **instances** using its most recent version of system and application configuration. Prior to starting applications, the *standby* performs the procedure for accelerated recovery of TCP connections, as described in the **dtproduct** command.

The *active* tracks the status of the other DT-6061, and remembers the most recent poll from the *standby*. If polls from the *standby* are overdue, the *active* initiates polling at a low frequency.

The physical proximity of the two DT-6061's must be near enough on the LAN to provide a reasonably reliable connection between the two. On the other hand, it is wise to avoid having them on the same power line.

If the network connection between the *active* and *standby* is broken, the network becomes partitioned; the *standby* can not receive responses from the *active*, and thus the *standby* will take over as *active* (at least on the portion of the network where it remains connected). When the connection is reestablished, the two *actives* soon see each other (because *actives* initiate polling when the *standby* has not been seen) and respond to this impermissible situation by rebooting. Upon reboot, they negotiate to choose a new *active* and *standby*.

Two new configuration objects are required for the high availability option: **ippublic** and **ipother**. When a DT-6061 is configured for high availability, the **vfy mod** command shows whether this DT-6061 is the *active* or *standby*.

### 10.1 MODULE CONFIGURATION FOR HIGH AVAILABILITY

**Syntax:** **ippublic** <ipaddress>  
**ipother** <ipaddress>

Before configuring a pair of DT-6061s for high availability, they should each be configured with their own **ipaddr**, and with the appropriate **gateway**, and **submask**. The two DT-6061's must



have the same gateway and must be on the same subnetwork, which is to say the IP addresses of the two DT-6061s must have the same network ID.

High availability configuration requires choosing a third IP address, **ippublic**, in the same network as the two DT-6061's. This *public* address is the one to be advertised to all other network elements that might be configured to interoperate with this DT-6061 pair. High availability is enabled by configuring each DT-6061 with both **ipother** (the **ipaddr** of the other DT-6061) and **ippublic**. Each DT-6061 must then be rebooted to enter high availability mode.

If it is necessary to move the DT-6061 pair to a different subnetwork, first *remove* the configuration for redundancy, setting both **ippublic** and **ipother** to *none*, then change the **ipaddrs** of both DT-6061s (see also **gateway** in section 6.7), then reconfigure the **ippublic** and **ipother** for the new subnetwork, then reboot.

**Syntax: ippublic none  
ipother none**

To *remove* the redundancy configuration for both DT-6061s, set both **ippublic** and **ipother** to *none* then reboot:

## 10.2 APPLICATION INSTALLATION AND OPERATIONAL CONFIGURATION

Software installation (**install**) and assignment of software to instances (**app**) are performed as usual on the *active* DT-6061. Application and platform software must be individually registered on both DT-6061s. The following describes how those operations are managed on the *standby*.

### 10.2.1 APPLICATION INSTALLATION

The recommended procedure for installing new application or platform software is to use the **install** command on both the *active* and *standby* DT-6061. It doesn't matter which DT-6061 is updated first. As with the standalone DT-6061, when performing **install** on the *active*, the affected instances are automatically restarted.

As an alternative, the *standby* may also receive newly installed software from the *active* DT-6061 by using the **stbyupd** command at the *standby* console:

**Syntax: stbyupd os | sw**

The **sw** option updates application software. If the *active* has been updated with platform software, **stbyupd os** should be performed on the *standby* followed by **stbyupd sw**. In order for the **stbyupd sw** or **os** commands to work properly, the current time and date must be set properly using **date** and **timezone** [6.4] on the *active* controller. If it is not convenient or feasible to check and maintain the current time, then it is recommended to use **install** to keep the *standby* software in sync with the *active*. The release notes for some platform updates *may* countermand use of **stbyupd os**, in which case the normal install procedures should be performed on the *standby*.

Following installation or the alternative **stbyupd** procedure, the software must be registered on both the *standby* and the *active*.

### 10.2.2 OPERATIONAL CONFIGURATION

The assignment of application types to instances, i.e. the **app** command, can only be performed on the *active* system console. After using **app** to change configuration on the *active*, the *standby* can have its configuration updated by entering **stbyupd cfg** on the *standby* console. This will also



update the *standby* with all current application configuration associated with the **instances** on the *active* DT-6061.

### 10.3 AUTOMATIC STANDBY CONFIGURATION UPDATE

The redundant pair of DT-6061's can be configured to perform a periodic automatic update of configuration from the *active* to the *standby*. This is the most convenient way to propagate configuration changes from the *active* system console and from the **instances** running on the *active* into the *standby* DT-6061.

The update period is configurable by the user at the *active* system console:

**Syntax:** `stby cfg SCAN=<minutes> AGE=<minutes>`

The SCAN period determines how often the *standby* polls the *active*, and the AGE value determines how old files must be before they will be automatically copied to the *standby*. Thus when the *active* fails and switches over, any configuration changes that were made in the last "AGE" minutes will not be present on the *standby*.

On the *active* system console, the **vfy stby** command displays the current status of the *standby* DT-6061 and the configured values for SCAN and AGE.

As part of automatic configuration update, when the *standby* detects that a new application has been configured, but the software for that application is not resident on the *standby*, it copies the software from the *active*, provided that application has been properly installed on the *active*. This is the *only* case where software is automatically updated on the *standby*. Otherwise, the procedures given in section 11.2.1 above must be performed in order to insure that the version of the application software used for a particular application is the same on the *standby* as it is on the *active* DT-6061. Either way, registration of the application on the *standby* *still* needs to be done manually.

The SCAN procedure requires quite a bit of overhead on the *active* DT-6061, so it is recommended to make the SCAN period as long as the user's requirements will tolerate (e.g. 60 minutes). The AGE delay prevents potentially fault-inducing configuration changes from propagating to the *standby*. If the *active* survives AGE minutes of operations with the new configuration, that configuration is propagated to the *standby*.

It is not necessary to maintain the current time and date for automatic update to work properly. However, if **date** and **timezone** [6.4] are used, an improper setting could cause undue delays in automatic update. In particular, manually setting the date backwards by a large amount while an update is pending could cause an update delay of the same amount.

### 10.4 SWITCHOVER

**Syntax:** `switchover`

The **switchover** command, available only at the *active* console, forces a switchover to the *standby* DT-6061. The former *active* DT-6061 then reboots into *standby* mode. The **reboot** command at the *active* console simply reboots the *active* DT-6061 back to *active* mode, while the *standby* DT-6061 allows a little extra time for the *active* to complete the **reboot**.

### 10.5 COMMANDS NOT AVAILABLE IN STANDBY MODE

The following commands are not allowed at the *standby* console:

- app
- backup
- banner





- cug
- dns
- dtproduct
- hosts
- remove/restore
- restart
- retrieve
- rstapp
- stby
- snmp
- switchover
- timeout
- timezone

The **help** command at the *standby* console shows only the commands that ARE allowed. The **date** command at the *standby* console may not be used to change the date. As described above, the *standby* receives the current time from the *active*.



## 11 APPENDIX A: BRIEF DESCRIPTIONS OF DT-6061 APPLICATIONS

### 11.1 IP-BHI

The IP-BHI (BiSync Host Interface) application allows a legacy BiSync host to connect to an IP network (using a DT-4000). Each instance of the application emulates one BiSync line with up to 32 cluster controllers and up to 32 logical devices per cluster controller. A DT-6061 can support up to 30 instances of the IP-BHI application; so one DT-6061 embedded network processor can support 30 host lines.

This application can also be used to replace the functionality of SYNC8 module used in BHIM mode in a BNS/Datakit® network.

### 11.2 TN3270

The TN3270 application will allow a PC that is connected to an IP network and has a 3270 client application installed within it to access a BiSync host via either the BNS/Datakit Sync8 module in BHIM mode or the IP-BHI application described above.

This application can also be used to replace the functionality of the Datakit TN3270 DKAP application.

### 11.3 IP-B2APRT

The BiSync to ASCII printer application, IP-B2APRT, allows a BiSync host to print to an ASCII printer located on an IP network. This application is usually used in conjunction with the TN3270, IP-BHI, and IP-SPOOL applications.

This application can also be used to replace the functionality of the Datakit A2B DKAP application.

### 11.4 IP-E2A

The IP-E2A application replaces the legacy E2A module function in a Datakit network. It provides the digital bridge function, thus eliminating the need for the high maintenance and difficult-to-fix digital networks used prevalently with E-telemetry. It can be used in an IP network to provide a gateway for E2 and E2A network elements to OS host applications.

### 11.5 IP-FANOUT

The IP-FANOUT application allows up to 504 IP-GATES (either standalone or those which are an embedded function of a DT-4000) to communicate over a WAN network.

### 11.6 IP-LTCN

The IP-LTCN application is a migration solution for all variants of the Loop Test Communications Environments. It is used in conjunction with an Internet Protocol conversion of the Tollgrade LoopCare® product. The LoopCare product is the evolution of the Lucent Technologies MLT operations system.

---

® Datakit is a registered trademark of Lucent Technologies, Inc, licensed to Datatek Applications, Inc., a company independent of Lucent Technologies, Inc.

® LoopCare is a registered trademark of Tollgrade, Inc.



The IP-LTCN can replace a DCN in its entirety. This eliminates a maintenance risk since the DCN is manufacturer discontinued, with a consequential shortage of spare parts. A side benefit is that the IP-LTCN will free up the 120 square feet of conditioned floor space taken up by the DCN.

The IP-LTCN supports an enhanced TCP/IP interface to the LoopCare host. This allows connectivity with the LoopCare hosts using an IP infrastructure. The new LoopCare hosts do not have either a BX.25 interface to communicate with a DCN, or a fiber interface to communicate with a BNS/Datakit node. Instead, the LoopCare hosts have implemented a direct TCP/IP (with RFC 854+885+1123 encapsulation) interface to the IP-LTCN application.

The IP-LTCN application relies on the DT-6061 redundancy infrastructure for its reliability. The DT-6061 infrastructure provides 100% reliability through a 1+1 spare arrangement. The IP-LTCN implements both the "A" and "B" cabinets of the DCN, and all interface paths of the LCN. Consequently, the IP-LTCN is the most reliable solution possible for the loop test environment.

## 11.7 X25PAD

The X25PAD Application eliminates the need, complexity, and expense of specialized X.25 equipment in an IP network. The X25PAD Application allows an X.25/BX.25 device, such as a network element's console, to be connected to a synchronous port anywhere in the IP network. Examples are a port on a DT-4XXX or DT-2020 connected SAM, or an endpoint such as a SAM port in a BNS/Datakit network that connects to an IP network through a UMI module. X25PAD Application connections to virtual circuits on an X.25/BX.25 port are provided with a telnet TCP/IP connection and support both X.3 PAD and Pass-Through functionality on a per virtual circuit basis.

This application can also be used to replace the functionality of the Datakit X25 Pass-Through DKAP application.

## 11.8 IP-SLM

The IP-SLM application performs a software emulation of the BNS BX.25 interface (also known as the Synchronous Line Module (SLM)). The IP-SLM application allows low cost and high-density access points to terminate BX.25 interfaces common to Network Elements. The result is significant cost savings, and administrative simplification. The IP-SLM is also a migration vehicle, which enables the reduction or elimination of proprietary network components for the Network Element support functionality.

Recent changes in requirements by Telcordia Technologies for the volume of information collected from network elements have generated the need for higher speed BX.25 links. The most cost effective way to meet this need is with the IP-SLM application. Synchronous line speeds as high as 56 Kbps are supported.

## 11.9 IP-SPOOL

Many existing networking equipment consoles have the need for a dedicated print device. Similarly, legacy hosts and their applications typically assume a dedicated printer of which their use is exclusive. Many of these print devices are under-utilized, but require considerable periodic maintenance. Finally, the physical location of these printers may not be convenient, due to requirements implied by their legacy interfaces.

The IP-SPOOL application provides print spooling per RFC1179 to a shared print resource. It provides the ability to route print output to a shared printer anywhere in the IP network. It reduces



the overall cost associated with legacy interfaces by eliminating underutilized, dedicated equipment and its maintenance.

One DT-6061, running the IP-SPOOL application, supports up to 30 units that can share a common physical printer. The actual number is dependent on the utilization of the previously dedicated printers and the print speed of the shared printer. More than one DT-6061 could share the same physical printer also.

## 11.10 IP-VCON

There is a need to deal with the multitude of *console* connections in modern networks. These consoles may be physical consoles, or logical connections via (B)X.25 logical channel connections, Telnet connections, etc. There is a need to monitor those consoles autonomously and unobtrusively.

The IP-VCON application takes over the mundane task of monitoring a large and various set of consoles autonomously, collecting the most important console messages, and intervening automatically when necessary according to user-defined scripts. When an administrator wishes to access any device console directly, the IP-VCON application provides the necessary connectivity without interrupting the monitoring function.

The IP-VCON application attaches via a TCP connection to any *console*, whether network based or physical. The protocol used by the console is unimportant. The application will monitor that console, take action on exceptions as needed, and provide a single means to access every console on and off the network. The application is eternally vigilant and will notify when a *console* indicates a problem situation. It can even determine an outage of equipment and cause appropriate actions to be executed.

The IP-VCON application can monitor and administer up to 768 consoles.

Via the application, each console may be monitored and administered by multiple human network administrators. Sixty-four administrators may be defined, and sixteen of these can be simultaneously accessing the application. Each administrator has its own login and password for access to the application. Each administrator has his its own list of consoles to which he/she is permitted access. As stated above, even if no administrators are logged onto the application, the application keeps track of what is happening for later use by an administrator and may take appropriate action in the interim.

## 11.11 ONSITE

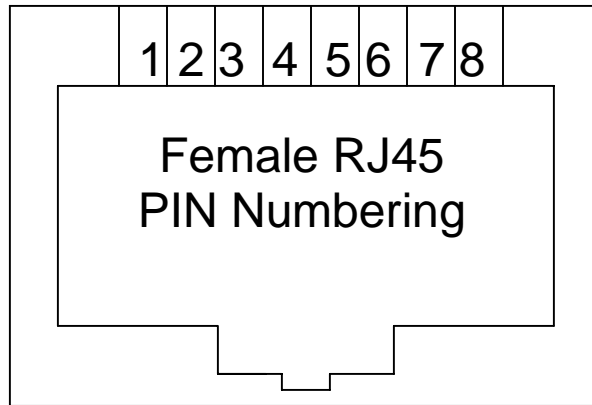
The ONSITE application for the DT-6061 Network Processor provides a connection and monitoring service for DT-9001 devices. This application translates DT-9001 temperature reports or contact state changes into text-based messages to be delivered to IP-connected host machines.

Multiple DT-9001 devices with multiple contacts can be monitored, and multiple hosts may receive messages as result of a single event from a DT-9001. Hosts, whose connections are configured and in service will receive text messages, subject to the condition of their TCP connection.

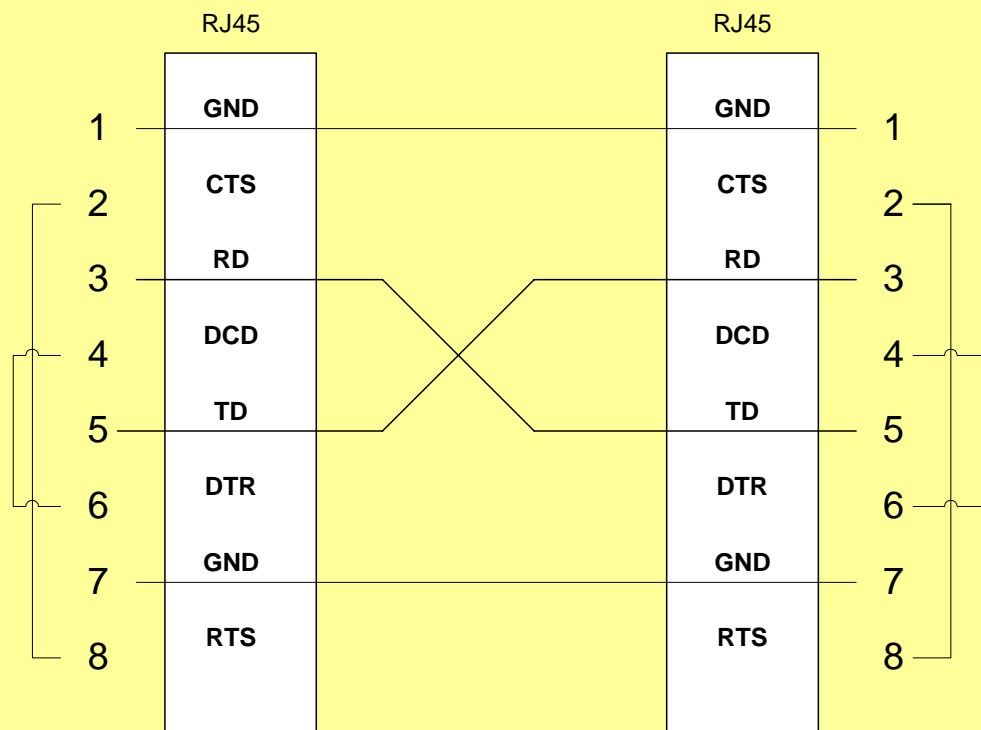


## 12 APPENDIX B: CONSOLE CABLE WIRING DIAGRAMS

Console Cable Pin Assignment



Console Cable - SPECIAL WIRING (DT-Products)

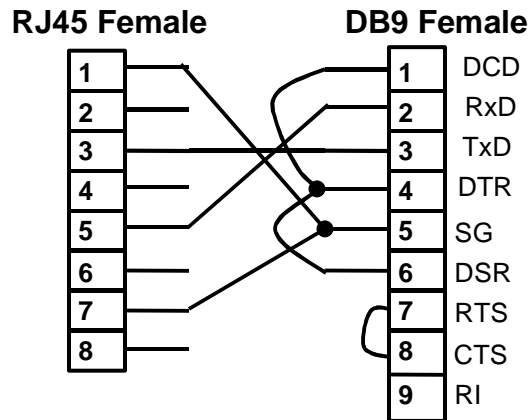
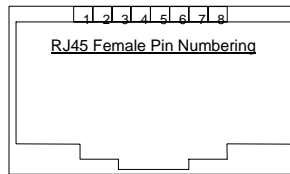


## 12.1 THE DB9 CONSOLE ADAPTER

Some Personal Computers use a 9-pin DB9 interface for serial communications. The terminal emulation programs may require certain lead status. Since console connections are generally implemented as three wire interfaces (i.e. RxD, TxD, and SG); this may pose a problem for the terminal emulation programs.

Below is depicted the wiring of a DB9 adapter which eliminates the problems associated with these terminal emulation programs. It is used with a standard **straight** category 5 RJ45 cable.

### DB9 Async DCE Console Adapter



Note: This cable for use on console ports only.

Use with a straight CAT-5 RJ45 cable.

#### 9-Pin Console Adapter Wiring Diagram



## 13 APPENDIX C: TCP/UDP PORT NUMBERS

The DT-6061 uses the following TCP and UDP ports. These ports cannot be changed by configuring them.

Protocol	Port Number	Purpose
UDP	161	Used by SNMP managers to set MIB values or get MIB data from the DT-6061
UDP	162	Used by the DT-6061 SNMP agent for sending Traps to the designated SNMP manager
TCP	1024-5000	Ports dynamically assigned by the DT-6061 for outgoing calls. Do NOT use or assign these ports.
UDP	9998	Active/Standby communication port
UDP	9999	Dtproduct - obsolete
TCP	10001-10030	The telnet console port for the application installed in instances 1 through 30 in a DT-6061.
TCP	30000-35800	Ports assigned in configuring various DT-6XXX applications



## 14 APPENDIX D: THE DISPLAY MEASUREMENTS (DM) REPORT

The DT-6061 platform maintains measurements of network activity as an aggregate of all the application instances. The measurements are grouped into four categories: IP, TCP, UDP, and ICMP. Many of these measurements are related to variables included in MIB II - Management Information Base for Network Management of TCP/IP-based Internets (RFC1213).

In the following report samples, the identifiers indicated in red have been added to reference individual measurements. Information is then provided (in some cases from RFC1213) to describe each measurement. Measurements that do not have identifiers (mainly in the TCP section) are those that would be helpful only to developers who are diagnosing problems, but would not be useful to end users on their own.

Please note that the measurements shown in the following report samples may not be representative of any particular customer's installation; i.e., the presence of 0's (or non-0's) is not necessarily an indication of what should be expected.

### 14.1 IP

```
<DT-6061> dm ip
```

```
ip:
```

```
IP 1 2488798 total packets received
IP 2 0 bad header checksums
IP 2 0 with size smaller than minimum
IP 2 0 with data size < data length
IP 2 0 with header length < data size
IP 2 0 with data length < header length
IP 2 0 with bad options
IP 2 0 with incorrect version number
IP 3 0 fragments received
IP 4 0 fragments dropped (dup or out of space)
IP 4 0 fragments dropped after timeout
IP 5 0 packets reassembled ok
IP 6 2414429 packets for this host
IP 7 74369 packets for unknown/unsupported protocol
IP 8 0 packets forwarded
IP 9 0 packets not forwardable
    0 redirects sent
IP 10 2186773 packets sent from this host
    0 packets sent with fabricated ip header
IP 11 0 output packets dropped due to no bufs, etc.
IP 12 0 output packets discarded due to no route
IP 13 0 output datagrams fragmented
IP 14 0 fragments created
IP 15 0 datagrams that can't be fragmented
```





**Measurement descriptions:**

IP 1: The total number of input datagrams received from interfaces, including those received in error.

IP 2: The number of input datagrams discarded due to errors in their IP headers, broken down by bad checksums, version number mismatch, other format errors, time-to-live exceeded, errors discovered in processing their IP options, etc.

IP 3: The number of IP fragments received which needed to be reassembled at this entity.

IP 4: The number of failures detected by the IP re-assembly algorithm (for whatever reason: timed out, errors, etc).

IP 5: The number of IP datagrams successfully re-assembled.

IP 6: The total number of input datagrams successfully delivered to IP user-protocols (including ICMP).

IP 7: The number of locally-addressed datagrams received successfully but discarded because of an unknown or unsupported protocol.

IP 8: The number of input datagrams for which this entity was not their final IP destination, as a result of which an attempt was made to find a route to forward them to that final destination. In entities that do not act as IP Gateways, this counter will include only those packets that were Source-Routed via this entity, and the Source-Route option processing was successful.

IP 9: The number of input datagrams discarded because the IP address in their IP header's destination field was not a valid address to be received at this entity. This count includes invalid addresses (e.g., 0.0.0.0) and addresses of unsupported Classes (e.g., Class E). For entities that are not IP Gateways and therefore do not forward datagrams, this counter includes datagrams discarded because the destination address was not a local address.

IP 10: The total number of IP datagrams that local IP user-protocols (including ICMP) supplied to IP in requests for transmission. Note that this counter does not include any datagrams counted in IP 8.

IP 11: The number of output IP datagrams for which no problem was encountered to prevent their transmission to their destination, but which were discarded (e.g., for lack of buffer space). Note that this counter would include datagrams counted in IP 8 if any such packets met this (discretionary) discard criterion.



IP 12: The number of IP datagrams discarded because no route could be found to transmit them to their destination. Note that this counter includes any packets counted in IP 8 which meet this 'no-route' criterion. Note that this includes any datagrams that a host cannot route because all of its default gateways are down.

IP 13: The number of IP datagrams that have been successfully fragmented at this entity.

IP 14: The number of IP datagram fragments that have been generated as a result of fragmentation at this entity.

IP 15: The number of IP datagrams that have been discarded because they needed to be fragmented at this entity but could not be, e.g., because their Don't Fragment flag was set.



## 14.2 TCP

<DT-6061> **dm tcp**

tcp:

```

TCP 1 1215000 packets sent
        74247 data packets (963479 bytes)
TCP 2 46 data packets (650 bytes) retransmitted
        0 resends initiated by MTU discovery
        1021108 ack-only packets (905646 delayed)
        0 URG only packets
        0 window probe packets
        68 rcv window update packets
        119724 control packets
TCP 3 2067762 packets received
        114483 acks (for 983715 bytes)
        81356 duplicate acks
        0 acks for unsent data
        1656659 packets (46725796 bytes) received in-sequence
        68313 completely duplicate packets (12111 bytes)
        0 old duplicate packets
        0 packets with some dup. data (0 bytes duped)
        7510 out-of-order packets (443 bytes)
        0 packets (0 bytes) of data after window
        0 window probes
        172 xmt window update packets
        0 packets received after close
        0 discarded for bad checksums
        0 discarded for bad header offset fields
        0 discarded because packet too short
TCP 4 99112 connection requests
TCP 5 7974 connection accepts
TCP 6 3 bad connection attempts
        50 listen queue overflows
TCP 7 20624 connections established (including accepts)
TCP 8 113169 connections closed (including 450 drops)
        51 connections updated cached RTT on close
        51 connections updated cached RTT variance on close
        8 connections updated cached ssthresh on close
TCP 9 86448 embryonic connections dropped
TCP 10 114403 segments updated rtt (of 200957 attempts)
TCP 11 19433 retransmit timeouts
TCP 12 2 connections dropped by rexmit timeout

```



```
TCP 13 0 persist timeouts
TCP 14 0 connections dropped by persist timeout
TCP 15 10223 keepalive timeouts
      0 keepalive probes sent
TCP 16 9630 connections dropped by keepalive
      22427 correct ACK header predictions
      1616588 correct data packet header predictions
```

**Measurement descriptions:**

TCP 1: The total number of segments sent, including those on current connections but excluding those containing only retransmitted octets.

TCP 2: The total number of segments retransmitted – that is, the number of TCP segments transmitted containing one or more previously transmitted octets.

TCP 3: The total number of segments received, including those received in error. This count includes segments received on currently established connections.

TCP 4: The number of times TCP connections have made a direct transition to the SYN-SENT state from the CLOSED state.

TCP 5: The number of times TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state.

TCP 6: The number of times TCP connections have made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state.

TCP 7: The number of TCP connections reaching the ESTABLISHED state.

TCP 8: The number of TCP connections or connection attempts that have been closed. Drops are closes initiated at the local end of the connection.

TCP 9: The number of connections closed that never reached ESTABLISHED state.



TCP 10: The count of window size updates based on round trip estimate.

TCP 11: The number of time expirations while awaiting an expected ACK.

TCP 12: The number of connections dropped because of the above timeout

TCP 13: The number of time expirations while waiting for window to open

TCP 14: The number of connections dropped because of the above timeout

TCP 15: The number of keep-alive timer expirations on idle connections

TCP 16: The number of connections dropped because of the above timeout

## 14.3 UDP

<DT-6061> **dm udp**

udp:

```
      834716 datagrams received
UDP 1  0 with incomplete header
UDP 1  0 with bad data length field
UDP 1  0 with bad checksum
UDP 2  1 dropped due to no socket
UDP 2 189454 broadcast/multicast datagrams dropped due to no socket
UDP 1  0 dropped due to full socket buffers
      645262 not for hashed pcb
UDP 3 645261 delivered
UDP 4 570930 datagrams output
```

### Measurement descriptions:

UDP 1: The number of received UDP datagrams that could not be delivered, broken down by reasons other than the lack of an application at the destination port.

UDP 2: The number of received UDP datagrams for which there was no application at the destination port.

UDP 3: The total number of UDP datagrams delivered to UDP users.

UDP 4: The total number of UDP datagrams sent from this entity.



## 14.4 ICMP

Note - within the output and input histograms embedded in this report, only parameters with non-zero values will be printed.

```
<DT-6061> dm icmp
icmp:
ICMP 1 1 call to icmp_error
        0 errors not generated 'cuz old message was icmp
        Output histogram:
ICMP 2      echo reply: 8
ICMP 3      destination unreachable: 1
ICMP 4      source quench: 1
ICMP 5      routing redirect: 2
ICMP 6      echo: 3
ICMP 7      time exceeded: 1
ICMP 8      parameter problem: 1
ICMP 9      time stamp: 2
ICMP 10     time stamp reply: 1
ICMP 11     address mask request: 1
ICMP 12     address mask reply: 1
        0 messages with bad code fields
        0 messages < minimum length
        0 bad checksums
        0 messages with bad length
        Input histogram:
ICMP 13     echo reply: 4
ICMP 14     destination unreachable: 74369
ICMP 15     source quench: 1
ICMP 16     routing redirect: 1
ICMP 17     echo: 8
ICMP 18     time exceeded: 1
ICMP 19     parameter problem: 1
ICMP 20     time stamp: 2
ICMP 21     time stamp reply: 2
ICMP 22     address mask request: 1
ICMP 23     address mask reply: 1
        8 message responses generated
```



**Measurement descriptions:**

ICMP 1: The number of ICMP messages which the entity received but determined as having ICMP-specific errors (bad ICMP checksums, bad length, etc.).

ICMP 2: The number of ICMP Echo Reply messages sent.

ICMP 3: The number of ICMP Destination Unreachable messages sent.

ICMP 4: The number of ICMP Source Quench messages sent.

ICMP 5: The number of ICMP Redirect messages sent.

ICMP 6: The number of ICMP Echo (request) messages sent.

ICMP 7: The number of ICMP Time Exceeded messages sent.

ICMP 8: The number of ICMP Parameter Problem messages sent.

ICMP 9: The number of ICMP Timestamp (request) messages sent.

ICMP 10: The number of ICMP Timestamp Reply messages sent.

ICMP 11: The number of ICMP Address Mask Request messages sent.

ICMP 12: The number of ICMP Address Mask Reply messages sent.

ICMP 13: The number of ICMP Echo Reply messages received.

ICMP 14: The number of ICMP Destination Unreachable messages received.

ICMP 15: The number of ICMP Source Quench messages received.

ICMP 16: The number of ICMP Redirect messages received.

ICMP 17: The number of ICMP Echo (request) messages received.

ICMP 18: The number of ICMP Time Exceeded messages received.

ICMP 19: The number of ICMP Parameter Problem messages received.

ICMP 20: The number of ICMP Timestamp (request) messages received.

ICMP 21: The number of ICMP Timestamp Reply messages received.

ICMP 22: The number of ICMP Address Mask Request messages received.

ICMP 23: The number of ICMP Address Mask Reply messages received.





## 15 APPENDIX E: DT-6061 SPECIFICATIONS

*Note: Shielded cables must be used in order to maintain compliance with EMC requirements.*

### 15.1 CON (CONSOLE)

A standard interface that uses binary data interchange between DTE and DCE. The RS-232C interface uses an RJ45 connector and operates at 9600 bits per second (bps).

### 15.2 RS-232/V.35

#### 15.2.1 RS-232C

A standard interface that uses binary data interchange between DTE and DCE. The RS232-C interface uses a 25-pin (DB25) connector and up to 21 signal leads, and operates at data rates from 75 to 19200 bits per second (bps).

#### 15.2.2 V.35

A standard interface used for interface and trunk modules. The V.35 interface uses a 34-pin connector and operates at data rates up to 2.048Mbps.

### 15.3 10-BASE-T LAN

Eight-pin, 10BaseT modular connector for a 10 Mbps baseband CSMA/CD local area network.

### 15.4 DSU

This interface uses an industry standard RJ48C connector. The interface may be used for T1 (1.544 MHz) or E1 (2.048 MHz)

### 15.5 PHYSICAL DIMENSIONS

DT-6061: L= 16.9 x W=8.77 x D=1.72

### 15.6 ENVIRONMENTAL OPERATING RANGE

Operating Temperature: 5° to 40°C (41°F to 104°F)

Operating Humidity: 5% to 85%

Altitude: From 60M (197) below sea level to 1800 m (5905 ft.) above sea level

### 15.7 POWER REQUIREMENTS

DT-6061 Operating Voltage: 5V @ 800 mA Nominal

Stand-alone AC to DC power supply: 115V @ 48mA Nominal  
115V @ 90 mA Maximum

Stand-alone DC power supply: 48V @ 104 mA Nominal  
48V @ 195 mA Maximum

Rack-mount DC (six units): 48V @ 624 mA Nominal  
48V @ 1.17 A Maximum



## 15.8 REGULATORY INFORMATION

### 15.8.1 DT-6061 STAND-ALONE

Safety:	UL, CSA, Low Voltage Directive 73/23/EEC
EMC:	FCC Part 15B Class A, ICES-003 Class A, EMC Directive 89/336/EEC
European Teleconnect	TTE Directive TBR 13
NEBS:	Level 3

To maintain compliance with the above-mentioned EMC standards, shielded cables must be used on all DT-6061 interface connections and the shields must make an electrical connection to the DT-6061's grounding system.

### 15.8.2 FCC PART 68 INFORMATION

The DT-6061 complies with FCC Part 68 of the FCC rules. On the bottom of the DT-6061 is a label that contains, among other information, the FCC registration number for the DT-6061. If requested, this information must be provided to the Telephone Company.

There is a single RJ45 style jack \*\*\* (labeled DSU)\*\*\* located on the front of the DT-6061 that has been tested as part of the registration process to FCC Part 68.

An FCC compliant modular jack is provided with the DT-6061. This jack is designed to be connected to the telephone network or premises wiring using a compatible modular plug, which is Part 68 compliant.

If the DT-6061 causes harm to the telephone network, the Telephone Company will notify you in advance that temporary discontinuance of service may be required. But if advance notice is not practical, the Telephone Company will notify you as soon as possible. Also, you will be advised of your right to file a complaint with the FCC if you believe it is necessary.

The Telephone Company may make changes in its facilities, equipment, operations or procedures that could affect the operation of the DT-6061. If this happens the Telephone Company will provide advance notice in order for you to make necessary modifications to maintain uninterrupted service.

If trouble is experienced please refer to the warranty section of this user's manual.

No repairs can be performed by you that go beyond the scope of the troubleshooting section of this user's manual.

### 15.8.3 INDUSTRY CANADA CS03 CERTIFICATION INFORMATION

NOTICE: The Industry Canada label identifies certified equipment. This certification means that the equipment meets the telecommunications network protective, operational and safety requirements as prescribed in the appropriate Terminal Equipment Technical Requirements document(s). The Department does not guarantee the equipment will operate to the user's satisfaction.

Before installing this equipment, users should ensure that it is permissible to be connected to the facilities of the local Telecommunications Company. The customer should be aware that compliance with the above conditions might not prevent degradation of service in some situations.



A representative designated by the supplier should coordinate repairs to certified equipment. Any repairs or alterations made by the user to this equipment, or equipment malfunctions, may give the telecommunications company cause to request the user to disconnect the equipment.

Users should ensure for their own protection that the electrical ground connections of the power utility, telephone lines and internal metallic water pipe system, if present, are connected together. This precaution may be particularly important in rural areas.

Caution: Users should not attempt to make such connections themselves, but should contact the appropriate electric inspection authority, or electrician, as appropriate.

## **15.8.4 NEBS**

### **GR-1089-CORE NEBS**

Section 6 DC Potential Difference  
Section 8 Corrosion Requirements

### **GR-1089-CORE NEBS Level 3**

Section 2 ESD  
Section 3.1, 3.2 EMI Emissions  
Section 3.3 Immunity  
Section 4 Lightning and AC Power Fault  
Section 5 Steady State Power Induction  
Section 7 Electrical Safety Analysis  
Section 9 Bonding and Grounding

### **GR-63-CORE NEBS**

Section 2 Spatial Requirements  
Section 4.1.3 Altitude  
Section 4.6 Acoustic Noise  
Section 4.7 Illumination Requirements

### **GR-63-CORE NEBS Level 3**

Section 4.1.1 Transportation and Storage  
Section 4.1.2 Operating Temperature and Humidity Criteria  
Section 4.2.3 Equipment Assembly Fire Test  
Section 4.3.1 Packaged Equipment Shock Criteria  
Section 4.3.2 Unpackaged Equipment Shock Criteria  
Section 4.4.1 Earthquake Environment and Criteria  
Section 4.4.3 Office Vibration Environment and Criteria  
Section 4.4.4 Transportation Vibration Criteria



## 16 HARDWARE WARRANTY

The warranty period for hardware shall be ninety (90) days from the date of shipment. Replacements and repairs are guaranteed for the longer of the remaining original warranty period or 30 days.

## 17 END-USER LICENSE AGREEMENT FOR SOFTWARE

This License Agreement ("License") is a legal contract between you and the manufacturer ("Manufacturer") of the system ("HARDWARE") with which you acquired software product(s) identified above ("SOFTWARE"). The SOFTWARE may include printed materials that accompany the SOFTWARE. Any software provided along with the SOFTWARE that is associated with a separate end-user license agreement is licensed to you under the terms of that license agreement. By installing, copying, downloading, accessing or otherwise using the SOFTWARE, you agree to be bound by the terms of this LICENSE. If you do not agree to the terms of this LICENSE, Manufacturer is unwilling to license the SOFTWARE to you. In such event, you may not use or copy the SOFTWARE, and you should promptly contact Manufacturer for instructions on return of the unused product(s) for a refund.

### 17.1 SOFTWARE LICENSE

You may only install and use one copy of the SOFTWARE on the HARDWARE (unless otherwise licensed by Manufacturer). The SOFTWARE may not be installed, accessed, displayed, run, shared or used concurrently on or from different computers, including a workstation, terminal or other digital electronic device ("Devices"). Notwithstanding the foregoing and except as otherwise provided below, any number of Devices may access or otherwise utilize the services of the SOFTWARE. You may not reverse engineer, decompile, or disassemble the SOFTWARE, except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation. The SOFTWARE is licensed as a single product. Its component parts may not be separated for use on more than one HARDWARE. The SOFTWARE is licensed with the HARDWARE as a single integrated product. The SOFTWARE may only be used with the HARDWARE as set forth in this LICENSE. You may not rent, lease or lend the SOFTWARE in any manner. You may permanently transfer all of your rights under this LICENSE only as part of a permanent sale or transfer of the HARDWARE, provided you retain no copies, you transfer all of the SOFTWARE (including all component parts, the media and printed materials, any upgrades, this LICENSE and, if applicable, the Certificate(s) of Authenticity), and the recipient agrees to the terms of this LICENSE. If the SOFTWARE is an upgrade, any transfer must also include all prior versions of the SOFTWARE. Without prejudice to any other rights, Manufacturer may terminate this LICENSE if you fail to comply with the terms and conditions of this LICENSE. In such event, you must destroy all copies of the SOFTWARE and all of its component parts.

### 17.2 INTELLECTUAL PROPERTY RIGHTS

The SOFTWARE is licensed, not sold to you. The SOFTWARE is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. You may not copy the printed materials accompanying the SOFTWARE. All title and intellectual property rights in and to the content which may be accessed through use of the SOFTWARE is the property of the respective content owner and may be protected by applicable copyright or other intellectual property laws and treaties. This LICENSE grants you no rights to use such content. All rights not expressly granted under this LICENSE are reserved Manufacturer and its licensors (if any).

### 17.3 SOFTWARE SUPPORT

SOFTWARE support is not provided by Manufacturer, or its affiliates or subsidiaries separate from the HARDWARE. For SOFTWARE support, please contact your supplier of the HARDWARE. Should you have any questions concerning this LICENSE, or if you desire to contact Manufacturer for any other reason, please refer to the address provided in the documentation for the HARDWARE.



## 17.4 EXPORT RESTRICTIONS

You agree that you will not export or re-export the SOFTWARE to any country, person, or entity subject to U.S. export restrictions. You specifically agree not to export or re-export the SOFTWARE: (i) to any country to which the U.S. has embargoed or restricted the export of goods or services, which as of March 1998 include, but are not necessarily limited to Cuba, Iran, Iraq, Libya, North Korea, Sudan and Syria, or to any national of any such country, wherever located, who intends to transmit or transport the products back to such country; (ii) to any person or entity who you know or have reason to know will utilize the SOFTWARE or portion thereof in the design, development or production of nuclear, chemical or biological weapons; or (iii) to any person or entity who has been prohibited from participating in U.S. export transactions by any federal agency of the U.S. government.

## 17.5 LIMITED WARRANTY

Manufacturer warrants that (a) the SOFTWARE will perform substantially in accordance with the accompanying written materials for a period of ninety (90) days from the date of shipment. Software support is limited to the hours of 8 AM to 5 PM ET Monday through Friday excluding Datatek-observed holidays.. Any implied warranties on the SOFTWARE are limited to ninety (90) days. Some states/jurisdictions do not allow limitations on duration of an implied warranty, so the above limitation may not apply to you.

Manufacturer's and its suppliers' entire liability and your exclusive remedy shall be, at Manufacturer's option, either (a) return of the price paid, or (b) repair or replacement of the SOFTWARE that does not meet this Limited Warranty and which is returned to Manufacturer with a copy of your receipt. This Limited Warranty is void if failure of the SOFTWARE has resulted from accident, abuse, or misapplication. Any replacement SOFTWARE will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer.

## 17.6 NO OTHER WARRANTIES

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, MANUFACTURER AND ITS SUPPLIERS DISCLAIM ALL OTHER WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT, WITH REGARD TO THE SOFTWARE AND THE ACCOMPANYING WRITTEN MATERIALS. THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHERS, WHICH VARY FROM STATE/JURISDICTION TO STATE/JURISDICTION.

## 17.7 SPECIAL PROVISIONS

The SOFTWARE and documentation are provided with RESTRICTED RIGHTS. Use, duplication, or disclosure by the United States Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and HARDWARE Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial HARDWARE Software-Restricted Rights at 48 CFR 52.227-19, as applicable. Manufacturer is Datatek Applications, Inc., 721 Rte. 202-206, Bridgewater, New Jersey 08807.

If you acquired the SOFTWARE in the United States of America, this Software License is governed by the laws of the State of New Jersey, excluding its choice of laws provisions. If you acquired the SOFTWARE outside the United States of America, local law may apply. This LICENSE constitutes the entire understanding and agreement between you and the Manufacturer in relation to the SOFTWARE and supercedes any and all prior or other communications, statements, documents, agreements or other information between the parties with respect to the subject matter hereof.

## 18 LIMITATION OF LIABILITY

To the maximum extent permitted by applicable law, in no event shall Manufacturer or its suppliers be liable for any damages whatsoever (including without limitation, special, incidental, consequential, or indirect damages for personal injury, loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use this product, even if Manufacturer has been advised of the possibility of such damages. In any case, Manufacturer's and its suppliers' entire liability under any provision of this License



shall be limited to the amount actually paid by you for the SOFTWARE and/or the HARDWARE. Because some states/jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

©Copyright 2000, 2002 TeleComp, Inc

©Copyright 2002, 2005 TeleComp Research and Development Corp.

©Copyright 2000, 2005 Datatek Applications, Inc.

All Rights Reserved

Printed in USA

