Configuration and Monitoring

Teldat Dm704-I

Legal Notice

Warranty

This publication is subject to change.

Teldat offers no warranty whatsoever for information contained in this manual.

Teldat is not liable for any direct, indirect, collateral, consequential or any other damage connected to the delivery, supply or use of this manual.

Table of Contents

I Related Documents

Teldat Dm211-I Quick Menu

Teldat Dm263-I Forwarder DEP

Teldat Dm296-I Telecontrol Module (MTC)

Teldat Dm689-I MTC+ Installation

Teldat Dm702-I TCP-IP Configuration

Teldat Dm703-I Frame Relay

Teldat Dm707-I X.25

Teldat Dm710-I PPP Interface

Teldat Dm711-I Frame Relay ISDN Backup Configuration

Teldat Dm712-I SNMP Agent

Teldat Dm713-I XOT Protocol

Teldat Dm715-I Bandwidth Reservation System

Teldat Dm722-I Telephony Over IP

Teldat Dm723-I DNS Client

Teldat Dm724-I FTP/sFTP Protocol

Teldat Dm725-I TVRP Protocol

Teldat Dm727-I Backup WAN Reroute

Teldat Dm728-I NTP Protocol

Teldat Dm732-I Dial Profile

Teldat Dm733-I RADIUS Protocol

Teldat Dm737-I HTTP Protocol

Teldat Dm738-I TELNET Protocol

Teldat Dm745-I Policy Routing

Teldat Dm749-I NSM

Teldat Dm751-I VLAN

Teldat Dm752-I Access Control

Teldat Dm753-I Syslog Client

Teldat Dm754-I NSLA

Teldat Dm757-I PCMCIA UMTS Interface

Teldat Dm765-I TFTP Protocol

Teldat Dm769-I STUN Protocol

Teldat Dm775-I VRF-Lite Facility

Teldat Dm780-I Prefix Lists

Teldat Dm784-I ISTUD Feature

Teldat Dm785-I DNS Updater

Teldat Dm786-I AFS

Teldat Dm787-I SSH Protocol

Teldat Dm789-I NETFLOW

Teldat Dm792-I Key Management

Teldat Dm793-I SPI Agent

Teldat Dm795-I Policy Map Class Map

Teldat Dm796-I RMON Feature

Teldat Dm797-I Dynamic Configuration Control

Teldat Dm800-I AAA Feature

Teldat Dm803-I Virtual Linux Interface (VLI)

Teldat Dm808-I IPv6 Access Control

Teldat Dm812-I GPS

Teldat Dm818-I ACT Feature

Teldat Dm819-I Wireless Network Management System

Teldat Dm820-I HotSpot Feature

Teldat Dm826-I CPE Wan Management Protocol (CWMP)

Teldat Dm831-I TTCP Feature

Chapter 1 Router console

1.1 Introduction

All of our routers use the same user interface regardless of the model. They differ in terms of the software of the protocols loaded on each device.

The information in this chapter is divided into the following sections:

- Local terminal and remote terminal.
- User interface.
- Description of the user interface.
- GESTCON process commands.

1.2 Local and remote terminal

The **Teldat Router** lets you use a local or remote terminal to configure and monitor its functions.

Local terminal

Local terminals are directly connected to the **Teldat Router** by an RS-232 serial cable. See your device's installation manual for more information.

Remote terminal

Remote terminals provide the same function as local terminals, except that you must use a local connection for initial configuration. Remote terminals connect to the **Teldat Router** via TELNET once the IP protocol has been enabled. For more information on enabling the IP protocol, see the following manual: *Teldat Dm702-I TCP-IP Configuration* .

Local or remote terminals allow you to access the **Teldat Router** to perform different processes. These are related to device configuration, monitoring and statistics, and you can also receive event messages. The following table outlines the various processes:

P 1 (GESTCON): This is the console management process. It is the starting point when starting a console session and provides access to the other processes.

P 2 (VISEVEN): This process allows you to view the events that occur in the system, from established connections to system errors. These events must be preprogrammed in process 4 (**CONFIG**) or process 3 (**MONITOR**) through the Event Logging System. See *Event Logging System ELS* on page 154 for more details.

P 3 (MONITOR): Allows you to **MONITOR** system status and any statistics gathered by the device.

P 4 (CONFIG): This process allows you to edit all the configuration parameters. From here you can create a full configuration without altering the device operation. You need to save the configuration and restart the device for the changes to take effect.

P 5 (RUNNING-CONFIG): This process allows you to change the device's active configuration. Changes made in this process take effect immediately, but any unsaved changes will be lost when the device is restarted.

Note

You can access these processes from the console by typing P 2, P 3, P 4 or P 5.

1.3 User interface

The following steps are the same for all **Teldat Router** models, regardless of the software loaded on the device.

- Connecting to the **Teldat Router**
- Executing a command
- User interface processes
- Access to processes

- Returning to the Management Console
- Obtaining help

1.3.1 Connecting to the Teldat Router

You can establish console sessions with the router either locally, via the RS-232 serial port, or remotely, via a TELNET session. The following sections describe these two access methods:

Local connection

Detailed information about the hardware and software configuration of the device and the system initialization progress are displayed at system startup. Once the boot phase is complete, the user is invited to start a console session by pressing any key.

The router's local connection is protected, so a username and password are required to access it. By default no users are registered. In this case, a username and password are not required when attempting to login and the following welcome dialog and Management Console prompt appear:

```
Teldat               (c)2001-2002
Router model XXXXX CPU MPC860      S/N: YYYY/YYYYY
1 LAN, 2 WAN Line , 2 ISDN Line
Operating System version: ZZZZZ
*
```

Where XXXXX is the specific router model.

To find out how to register a user, see the **user** command in *Teldat Router Configuration* on page 19. Registered users are prompted to enter a username and password. If authentication is accepted, the welcome dialog appears.

```
User: Root
Password:****
Teldat               (c)2001-2002
Router model XXXXX CPU MPC860      S/N: YYYY/YYYYY
1 LAN, 2 WAN Line , 2 ISDN Line
Operating System version: ZZZZZ

*
```

Where XXXXX is the specific router model, YYYY/YYYYY is the unit serial number, and ZZZZ is the OS version that is running.

If the user enters an invalid password, the following text appears:

```
User: Root
Password:******
Access denied
```

If the password is not correct, the console will be inaccessible. The application blocks the user for 1 minute after the maximum number of incorrect password attempts.

If a user is authenticated and an idle timeout is configured (see the **set** command in *Teldat Router Configuration* on page 19), the router will drop the connection if the user is idle for more time than the configured idle timeout value. The user must then reenter the password in order to gain access to the console again.

A user access level determines the type of processes and commands available.

The user access level is specified with a value from 0 to 15 and a mode, *default* or *strict*.

There are five predefined access levels in default mode:

NONE [0]: Does not allow access to the system.

EVENTS [1]: Accesses Console Management (P1) and Event Viewing (P2) but not to the **Ping**, **Telnet**, **Restart** or **Load** commands.

MONITOR [5]: Accesses Console Management (P1), Event Viewing (P2), and Monitoring (P3). It also allows you to run the **Ping** and **Telnet** commands, but not **Restart** or **Load**.

CONFIG [10]: Access to all processes and all standard commands.

ROOT [15]: As well as having access to all processes and standard commands, you have access to the user management commands themselves (which are explained later).

You must save the configuration (see the **save** command in *Teldat Router Configuration* on page 19) if you want users to continue to be able to log in after a device restart. Otherwise, their settings will be lost.

To manage users you use the **user** command. It allows you to add, delete, enable and disable users, and list and change access levels:

user name **password:**

Configures a user's password, creating it if one does not exist.

no user name**:**

Deletes a user from the user list. You can delete as many users as you want, but not the last root user if there are still some users registered in the system. In this case, you can only delete the other registered users (were you to actually delete the last root user without removing the other registered users, then you wouldn't be able to manage those users). You can delete the last root user once you have removed all the registered users. Then the system would no longer request a username and password to access the device because there would be no users left in the system.

user name **active**:

Allows you to enable users. You simply indicate the user name you want to enable.

user name no **active**:

Disables users. Disabling root users is not allowed.

list user:

Displays the list of registered users, their access level, and whether or not they are enabled.

Allows you to change a registered user's access level, but not for users with root level access.

The **user** command is described in greater detail in *Teldat Router Configuration* on page 19.

User management is compatible with the password specified with the **set password** command. Thus, if you update a device that has this password enabled, the device will continue to allow access when that password is entered while no user is registered.

> **Note**
>
> User management has priority over the device password. Thus, once users are registered and enabled (by default, users are enabled when they are registered), the old password will no longer be valid.

Remote connection

To connect to the **Teldat Router** by initiating a TELNET session on the host (the *host* is the system where the remote terminal resides), you need to provide the IP address of the device you want to connect to.

Example:

```
telnet  128.185.132.43
```

The **Teldat Router** acts as a *TELNET server*. The remote terminal acts as a *TELNET client*.

Once a TELNET session is established with the **Teldat Router**, if necessary, the user is required to enter a username and password to access the system. Once the user is authenticated, the welcome dialog appears:

```
User: Root
Password:****
Teldat               (c)2001-2002
Router model XXXXX CPU MPC860      S/N: YYYY/YYYYY
1 LAN, 2 WAN Line , 2 ISDN Line
Operating System version: ZZZZZ
*
```

Where XXXXX is the specific router model, YYYY/YYYY is the unit serial number, and ZZZZ is the OS version that is running.

Access control on the **Teldat Router** is similar to local mode access. If users have been defined and are enabled (they are enabled by default upon creation), they are prompted to enter a username and password to allow them to connect to the system. When the authentication is correct, the welcome dialog and prompt appear and the authenticated user's privileges will be available.

If the user enters an invalid password, the following text appears:

```
User: Root
Password:******
Access denied
```

Note

If the password is not entered within 20 seconds or is entered incorrectly three times in a row, the device will disconnect the TELNET session.

1.3.2 Running a command

Commands may be abbreviated as long as enough characters are used to distinguish them from other commands in the current menu.

Example:

If we type **u** in the menu with the **user**, **upload**, and **down** commands, we will get an error message telling us that we have entered an ambiguous command **(user** and **upload** both begin with the letter u). Typing **d** , *do*, **dow** or **down** will run the **down** command, typing **us** , **use** or **user** will run the **user** command, while **upl** , **uplo** , **uploa** or **upload** will run the **upload** command. Any other input will produce an error because no command will match what is typed.

To delete the last character(s) in the command line, use the backspace (**<-**) key.

To split a long command into several lines, type a backslash (****) at the end of the line.

Example:

```
Config>set \
Config>host\
Config>n MY_\
Config>HOST_N\AME
```

is equivalent to:

```
Config>set hostn MY_HOST_N\AME
```

Command history:

Release	Modification
11.00.05	This command option was introduced as of version 11.00.05.

1.3.3 User Interface Processes

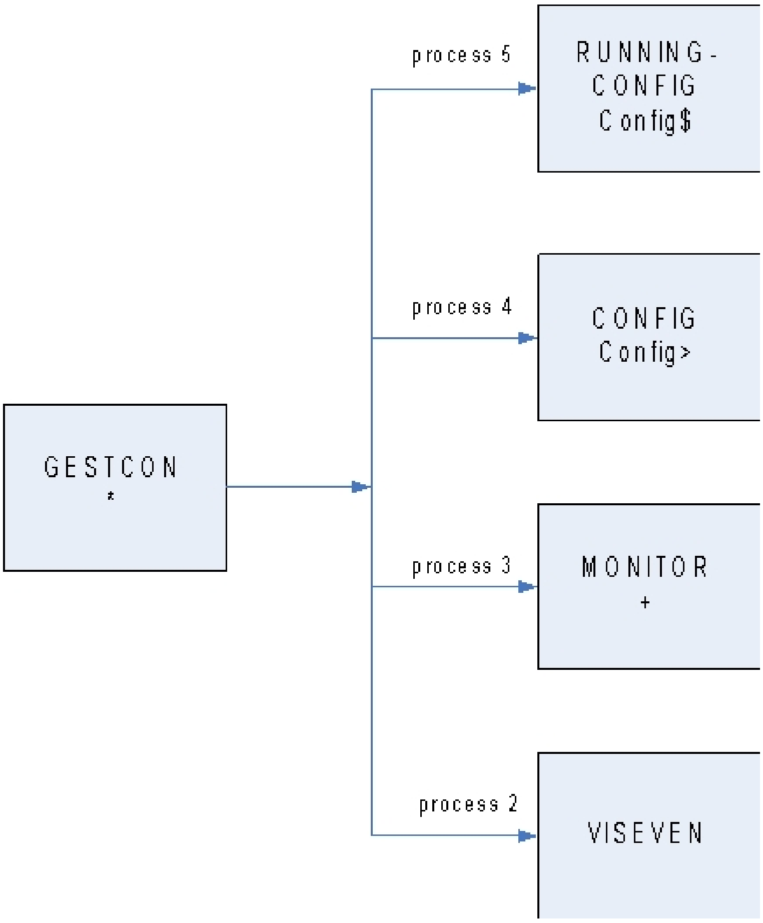
The user interface consists of several processes that are accessed through console sessions.

The most common processes are: GESTCON, MONITOR, CONFIG, RUNNING-CONFIG and VISEVEN. The following diagram shows the structure of the processes in the **Teldat Router**.

As you can see, each process has a different prompt. You can tell which process you are in by looking at the prompt.

The following list shows the prompts for the different processes:

Process	Prompt
GESTCON	*
MONITOR	+
CONFIG	*Config>*
RUNNING-CONFIG	*Config$*



The **Teldat Router** offers the possibility of personalizing the device by inserting a text string before the prompt. This text can be up to 8 characters long, and is taken from the name assigned to the device. To enter it, see the **set hostname** configuration command.

The following sections focus on each process:

GESTCON process

This is the Management Console and its mission is to provide access to the other processes.

MONITOR process

This allows the user to monitor the router's hardware/software status and statistics. It provides access to the protocol and interface menus. These menus allow the user to monitor configured protocols and other parameters.

CONFIG process

This allows you to configure various parameters, such as network addresses and events. It provides access to the protocol configuration environment to configure protocol parameters. From this process you can perform a complete device configuration, but the changes will not take effect until they are saved and the device restarted. Therefore, this process is used to modify the device's startup configuration.

RUNNING-CONFIG process

This provides access to the configuration of interfaces, protocols, etc. All parameters configured from here will take effect immediately, but unsaved changes will be lost when the device is restarted. Therefore, this process is used to dynamically modify the device's active configuration.

VISEVEN process

Receives system event messages and displays them on the terminal according to user selection criteria.

1.3.4 Connecting to a processes

Once the session is started, the Management Console prompt (*) appears on the console. This is the command prompt from which you communicate with the different processes. *Prompts* are symbols that identify processes:

To connect to a process you must:

(1) Find out the process identifier number. You can get this information by typing the **status** command at the asterisk (*) prompt.

(2) Type the process **pid** command, where pid is the number of the process to which you want to connect. For example, to configure the **Teldat Router**, you would type:

```
*process 4
User Configuration
Config>
```

However, there are a number of specific commands you can use to access the most common processes. The following table shows these commands:

Command	**Process**
monitor	Process 2: monitoring.
config	Process 4: startup configuration editor.
running-config	Process 5: active configuration.

1.3.5 Returning to the Management Console

To get back to the Management Console (*) prompt when you are finished with a process, such as CONFIG (Config> prompt) or MONITOR (+ prompt), press the escape character *Ctrl+p*. **You MUST ALWAYS RETURN to the Management Console BEFORE GOING TO ANOTHER PROCESS**. For example, if you are connected to the MONITOR process and you want to connect to the CONFIG process, you must press *Ctrl+p* to return to the Management Console prompt (*) first.

To end a TELNET session that you have initiated with another device from the **Teldat Router**, you can use the *Ctrl+s* escape character. This escape character forces the TELNET session initiated by the **Teldat Router** to close.

Note

To return to the Management Console, use the *Ctrl+p* escape character. To close a TELNET session, use *Ctrl+s*.

Example:

```
*config
User Configuration
Config>                                                          Press (Ctrl + p)
*
```

```
*monitor
Console Operator
+                                                                Press (Ctrl + p)
*
```

Note

The configuration and monitoring processes allow you to access other protocol configuration/monitoring menus. To return to a higher level configuration/monitoring process, use the **exit** command. To return to the Management Console, use the *Ctrl+p* escape character.

1.3.6 Getting help

At the command prompts in both the Management Console (*) and the configuration (**Config>** and **Config$**) and monitoring (**+**) processes, you can obtain help in the form of a listing of the commands available at that level.

You can also type a question mark (**?**) after a command or option to obtain a corresponding list of commands or options. In addition, you can press the tab key to automatically complete a command or option that is already perfectly defined.

Example:

```
Config>protocol ?
  arp     Access ARP protocol
  asrt    Access ASRT protocol
  bfd     Access BFD protocol
  bgp     Access BGP protocol
  dhcp    Access DHCP protocol
  dhcpv6  Access DHCPv6 protocol
  dls     Access DLS protocol
  dot1x   Access 802.1X protocol
  gw104   Access GW-104 protocol
  h323    Access H323 protocol
  igmp    Access IGMP protocol
  ip      Access IP protocol
  ipv6    Access IPv6 protocol
  l2tp    Access L2TP protocol
  mgcp    Access MGCP protocol
  msdp    Access MSDP protocol
  nhrp    Access NHRP protocol
  noe     Access NOE protocol
  ospf    Access OSPF protocol
  ospfv3  Access OSPFv3 protocol
  pim     Access PIM protocol
  rip     Access RIP protocol
  ripng   Access RIPNG protocol
  sccp    Access SCCP protocol
  sip     Access SIP protocol
  snmp    Access SNMP protocol
Config>protocol a?
  arp     Access ARP protocol
  asrt    Access ASRT protocol
Config>protocol a
```

1.4 GESTCON commands

The GESTCON process (P1) allows you to configure and monitor all the device's operating parameters. While we are in the GESTCON process, the **Teldat Router** is processing and forwarding data traffic. When the router is turned on and enters the GESTCON process, copyright information and an asterisk (*) appear on the locally connected terminal. This asterisk is the GESTCON prompt or main user interface that allows you to access other (second-level) processes. Most of the changes made to the **Teldat Router** operating parameters in the GESTCON process take effect immediately, without having to restart the device.

Within the GESTCON process there is a set of commands for checking the status of processes, monitoring interfaces and packet transfer, and configuring several parameters.

GESTCON process commands table

Commands	**Function**
MONITOR	Accesses the monitoring process.
CONFIG	Accesses the boot configuration editing process.
RUNNING-CONFIG	Accesses the active configuration editing process.
FLUSH	Clears all messages stored in the event buffer so far.
INTERCEPT	Allows you to change the process escape character.
LOAD	Reloads the application from flash memory.
LOGOUT	Terminates a Telnet connection established with the device.
PROCESS	Lets you access another device process and enable its commands.
RESTART	Restarts the device by re-reading the configuration.
SSH	Opens an SSH client connection to a remote device whose address is specified.
STATUS	Displays process names and identifiers.
TELNET<address>	Opens a Telnet client connection to a remote device whose address is specified.
VRF-TELNET <vrf> <address>	Opens a Telnet client connection to a remote client device whose address is specified in the specified VRF.

1.4.1 MONITOR

Accesses the monitoring process.

Syntax:

```
*monitor
```

Example:

```
*monitor
Console Operator
+
```

1.4.2 CONFIG

Accesses the startup configuration editing process.

Syntax:

```
*config
```

Example:

```
*config
Config>
```

1.4.3 RUNNING-CONFIG

Accesses the active configuration editing process.

Syntax:

```
*running-config
```

Example:

```
*running-config
Config$
```

1.4.4 FLUSH

Clears the event viewer (VISEVEN) output buffer of all events.

Syntax:

```
*flush
```

Example:

```
*flush
*
```

1.4.5 INTERCEPT

Allows you to change the process escape character. In the example below, the default escape character is changed from **Ctrl+u** to **Ctrl+p**.

Syntax:

```
*intercept
```

Example:

```
*intercept
Press the new escape key and then Enter:                    Press (Ctrl+u) and <¿>
Press the new escape key again and then enter:              Press (Ctrl+u) and <¿>
Escape key updated
*
```

Note

The escape key must not be a character that can be displayed.

1.4.6 LOAD

Allows you to load the application from flash memory.

Syntax:

```
* load <option>  [yes]
ACTIVATE
DEACTIVATE
IMMEDIATE
RACTIVATE
RDEACTIVATE
```

- **< option >** specifies the type of load you want to use.
- **yes** option can be used to bypass the systems confirmation question.

1.4.6.1 LOAD ACTIVATE

The **activate** option allows you to program the router to load the routing application at a certain time. The time is set in 24-hour format.

Syntax:

```
*load activate [yes]
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.

Example:

```
*load activate 17:21
Are you sure to reload the system at the configured time (Yes/No)? y
*
```

Command history:

Release	Modification
11.01.06	The "[yes]" option was introduced as of version 11.01.06

1.4.6.2 LOAD DEACTIVATE

The **deactivate** option overrides a programmed reload that has not completed. If there are no programmed reloads, an error message is displayed.

Syntax:

```
*load deactivate [yes]
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.

Example:

```
*load deactivate
Reload is timed at 20:00
Are you sure to cancel the timed reload(Yes/No)? y
Timed reload was canceled
*
```

Command history:

Release	Modification
11.01.06	The "[yes]" option was introduced as of version 11.01.06

1.4.6.3 LOAD IMMEDIATE

The **immediate** option reloads the application instantly.

Syntax:

```
*load immediate [yes]
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.

Example:

```
*load immediate
Are you sure to reload the device(Yes/No)? y
*
```

Command history:

Release	Modification
11.01.06	The "[yes]" option was introduced as of version 11.01.06

1.4.6.4 LOAD RACTIVATE

Allows the user to program the routing application program to restart at a certain time. The time is set in 24-hour format.

Syntax:

```
*load ractivate [yes]
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.

Example:

```
*load ractivate 17:26
Are you sure to restart the system at the configured time (Yes/No)? y
*
```

Command history:

Release	Modification
11.01.06	The "[yes]" option was introduced as of version 11.01.06.

1.4.6.5 LOAD RDEACTIVATE

Disables a programmed restart. If there are no programmed restarts, an error message is displayed.

Syntax:

```
*load rdeactivate [yes]
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.

Example:

```
*load rdeactivate
Reload is timed at 17:00
Are you sure to cancel the timed restart(Yes/No)? y
Timed restart was canceled
*
```

Command history:

Release	Modification
11.01.06	The "[yes]" option was introduced as of version 11.01.06.

1.4.7 LOGOUT

Terminates the Telnet connection to the router without using any Telnet client commands.

Syntax:

```
*logout
```

Example:

```
*logout
Do you wish to end telnet connection (Yes/No)?
```

1.4.8 PROCESS

Allows you to access other processes, such as MONITOR, VISEVEN, or CONFIG. After connecting to a new process, you can send specific commands or receive output from that process. To obtain the process identifier, use the **status** command. Once you are connected to another process, such as MONITOR, VISEVEN, or CONFIG, use the *Ctrl+p* escape character to return to the Management Console (GESTCON).

Syntax:

```
*process <pid>
```

- **< pid >** this is the identifier of the process whose console you want to access.

Example:

```
*process 4
User Configuration
Config>
```

When you are in any of the protocol menus (e.g., *Conf IP>* or *IP>*), use the **exit command** to return to a process menu.

1.4.9 RESTART

Restarts the **Teldat Router** without reloading the software. The router then:

- Sets the software counters to zero.
- Tests the connected networks.
- Deletes any routing tables.
- Discards all packets until the restart completes.
- Executes the current software.

> **Note**
>
> If this command is used on a remote terminal connection, the TELNET session is lost because all the processes on the device are restarted.

Syntax:

```
*restart [yes]
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.

Example:

```
*restart
Are you sure to restart the system(Yes/No)? y
  Done
Restarting. Please wait ..............................................
APP DATA DUMP.........................................
Running application
Flash configuration read
Parsing text mode configuration ...
Configuration parsed
Initializing
Press any key to get started
```

Command history:

Release	Modification
11.01.06	The "[yes]" option was introduced as of version 11.01.06.

1.4.10 SSH

Establishes an SSH connection to a remote device with a specified address. The address can be an IPv4 address or a domain name if you configure a DNS client.

Syntax:

```
*ssh [vrf <vrf>] <address> [login <login-name> | port <port> | version <1-2> | cipher <cipher> | <cr> ]
```

The **vrf** option allows you to specify the VRF that will be used to initiate the SSH session. If this option is not used, then the primary VRF is used.

You can specify the following parameters in SSH:

- **vrf < vrf >** specifies the VRF that will be used to initiate the SSH session.
- **< address >** specifies which IP address or device domain name to access via SSH.
- **login < login-name >** specifies the user ID to use for logging onto the remote networking device running the SSH server. When no user ID is specified, the current user ID is used by default.
- **port < port >** specifies the remote host's port number. The default is 22.
- **version < 1-2 >** specifies the SSH version to use.
- **cipher < cipher >** selects the cipher specification for encrypting the session.

Examples:

The following example will establish an SSH connection to the router with IP address 192.168.212.201 using the specified user ID:

```
Router2 *ssh 192.168.212.201 login mbejar
(Press Control T to come back to local router)
The authenticity of host '192.168.212.201 (192.168.212.201)' can't be established.
RSA1 key fingerprint is 59:b7:6e:d3:1e:85:95:b4:88:03:a7:2e:3d:35:41:c5.
Are you sure you want to continue connecting (Yes/No)? yes
mbejar@192.168.212.201's password: ******


Teldat                (c)2001-2015
Router model XXXXX CPU MPC860      S/N: YYYY/YYYYY

1 LAN, 2 WAN Line , 2 ISDN Line
CIT software version: ZZZZZ


Router1 *
```

The following example will establish an SSH connection to the router with IP address 192.168.212.201 without using the user ID (local *root* user).

```
Router2 *ssh 192.168.212.201
(Press Control T to come back to local router)
root@192.168.212.201's password: ******


Teldat                (c)2001-2015
Router model XXXXX CPU MPC860      S/N: YYYY/YYYYY

1 LAN, 2 WAN Line , 2 ISDN CIT software version: ZZZZZ


Router1 *
```

The following example will establish an SSH connection to the router with IP address 192.168.212.201 using the user ID and port 50 when the SSH server is NOT listening to that port and the connection cannot be established.

```
Router2 *ssh 192.168.212.201 login mbejar port 50
(Press Control T to come back to local router)
Connect to host 192.168.212.201 port 50: Connection reset by peer.
Router2 *
```

The following example will establish an SSH connection to the router with IP address 192.168.212.201 using the user ID and port 50 when the SSH server is listening to port 50 as well.

```
Router2 *ssh 192.168.212.201 login mbejar port 50
(Press Control T to come back to local router)
mbejar@192.168.212.201's password: ******


Teldat               (c)2001-2015
Router model XXXXX CPU MPC860      S/N: YYYY/YYYYY

1 LAN, 2 WAN Line , 2 ISDN Line
CIT software version: ZZZZZ


Router1 *
```

The following example will establish an SSH connection to the router at 192.168.212.201 forcing version 1 when the SSH server only supports SSHv2 and the connection cannot be established.

```
Router2 *ssh 192.168.212.201 login mbejar version 1
(Press Control T to come back to local router)
Protocol major versions differ: 1 vs. 2.
SSH connection closed.
Router2 *
```

The following example will establish an SSH connection to the router with hostname ares.id.teldat.com with the user ID.

```
Router2 *ssh ares.id.teldat.com login mbejar
(Press Control T to come back to local router)
The authenticity of host '192.168.212.7 (192.168.212.7)' can't be established.
RSA key fingerprint is b4:53:cc:47:3e:bc:c9:82:ca:0c:97:8e:4f:f4:6c:b6.
Are you sure you want to continue connecting (Yes/No)? yes
mbejar@192.168.212.7's password: ******
Linux ares 3.2.0-4-amd64 #1 SMP Debian 3.2.60-1+deb7u3 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Nov 20 08:31:24 2015 from 192.168.212.19
mbejar@ares:~$
```

The following example will establish an SSH VRF connection to the router with IP address 192.168.140.254 using a user ID.

```
Router2 *ssh vrf client 192.168.140.254 login malonso
Trying to connect...
(Press Control T to come back to local router)
malonso@192.168.140.254's password:
```

Command history:

Release	Modification
11.00.05	SSH client was introduced as of version 11.00.05.
11.01.00	SSH client was introduced as of version 11.01.00.
11.00.06	This command was modified as of version 11.00.06. "Ctrl T" is the new escape key.
11.01.02	This command was modified as of version 11.01.02. "Ctrl T" is the new escape key.
11.01.09	SSH VRF command was introduced as of version 11.01.08.

1.4.11 STATUS

This option allows you to find out process names and identifiers (PID).

Syntax:

```
*status
```

Example:

```
*status
System Processes:
 PID  NAME

  1   Main console
  2   Event viewer
  3   Monitor console
  4   Config console
  5   Running config console
  6   Telnet client
*
```

1.4.12 TELNET

Establishes a Telnet connection to a remote device with a specified address. You can use this command with IPv4/IPv6 addresses or with a domain name if you configure a DNS client.

Syntax:

```
*telnet
  [vrf <vrf>] <address> [source <address>] [port <port>]
  <cr>
```

The **vrf** option allows you to specify the VRF that will be used to initiate the Telnet session. If this option is not used, then the primary VRF is used.

If no parameter is entered (**telnet <cr>** option), all telnet parameters are requested and the primary VRF is used. In this case, the telnet source and destination addresses can only be IPv4.

The parameters that can be specified in Telnet are:

- **vrf <vrf>** specifies the VRF that will be used to initiate the Telnet session.
- **<address>** specifies the IP address or domain name of the device to access via Telnet.
- **source <address>** specifies the source IP address to use for Telnet. If a domain name is specified as the destination, the source address can only be IPv4.
- **port <port>** specifies the destination port to use for Telnet.

Examples:

The following example will telnet to the router with IP address 172.123.23.67:

```
*telnet 176.123.23.67
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

The following example will telnet to the router with IP address 172.24.78.92 using the source address 80.1.1.1 and port 6623.

```
FTP *telnet
Telnet destination []? 172.24.78.92
Telnet source [172.24.78.94]? 80.1.1.1
Telnet port [23]? 6623
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

The following example will telnet to the router at 172.24.78.92 using the source address 80.1.1.1 and port 6623, specifying the parameters with options.

```
FTP *telnet 172.24.78.92 source 80.1.1.1 port 6623
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

The following example will telnet to the router with IPv6 address 2001:db8:1::1 using the source address 2001:db8:1::2. Since it is an IPv6 destination, you must specify the parameters with options:

```
*telnet 2001:db8:1::1 source 2001:db8:1::2
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

The following example will telnet to the router with domain name router1.midominio.es:

```
FTP *telnet router1.midominio.es
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

The following example will telnet to the router at 172.24.78.92 using the router_aux VRF.

```
FTP *telnet vrf router_aux 172.24.78.92
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

Command history:

Release	Modification
11.01.09	IPv6 VRF support was introduced as of version 11.01.09.

1.4.13 VRF-TELNET

Establishes a Telnet connection to a remote device with a specified address in a particular VRF. The address can be a domain name if the DNS client is configured.

Syntax:

```
*telnet <vrf>
             <address> [source <address> | port <port> | <cr> ]
             <cr>
```

- **< vrf >** specifies the name of the VRF you want to use to initiate the Telnet session.

If no additional parameters are entered (**vrf-telnet <vrf> <cr>** option), you are prompted to enter all Telnet parameters.

The parameters that can be specified in Telnet are:

- **< address >** specifies the IP address or domain name of the device to access via Telnet.
- **source < address >** specifies the source IP address to use for Telnet.
- **port < port >** specifies the destination port to use for Telnet

Examples:

The following example will telnet to the router with IP address 172.123.23.67 in the VRF called *client*.

```
*vrf-telnet client 176.123.23.67
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

The following example will telnet to the router with IP address 172.24.78.92 in the VRF called *client* using the source address 80.1.1.1 and port 6623.

```
FTP *vrf-telnet
vrf tag []? client
Telnet destination []? 172.24.78.92
Telnet source [172.24.78.94]? 80.1.1.1
Telnet port [23]? 6623
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

The following example will telnet to the router at 172.24.78.92 in the VRF called *client* using the source address 80.1.1.1 on port 6623, specifying the parameters with options.

```
FTP *vrf-telnet client 172.24.78.92 source 80.1.1.1 port 6623
```

```
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

The following example will telnet to the router with the domain name router1.midominio.es in the VRF called *client*:

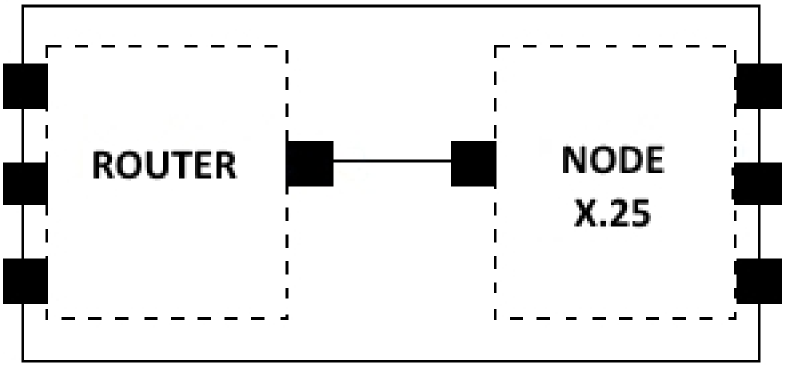
```
FTP *vrf-telnet client router1.midominio.es
Trying to connect...
(Press Control S to come back to local router)
Connection established
```

Chapter 2 Teldat Router Configuration

2.1 Introduction

From a functional point of view, the **Teldat Router** has two virtual devices:

- A router that performs the internetworking functions.
- A packet switch for packets coming from both the router and the X.25 and ISDN ports, when they carry X.25.



As you can see in the diagram, each virtual machine governs its own set of interfaces. It is therefore necessary to be able to accurately identify the different interfaces and to know whether an interface belongs to the router or to the node.

The way that you identify the interfaces in the **Teldat Router** configuration is by means of a name.

For physical interfaces, the name consists of a string of text followed by two numbers separated by a slash. The text indicates the type of interface (serial line, ISDN, etc.), the first of the numbers indicates the interface's location in the device (0 for the motherboard, 1 for the first pci extension, 2 for the second, etc.), and the second number indicates the number of occurrences of that interface type for a single location (serial line 0, 1, 2, etc.).

In user-added interfaces, the name consists of a text string that indicates the interface type and a number that is unique for that type of interface. Subinterfaces are an exception to this rule as their name consists of the base interface name followed by a period (**.**) and then by a number that is unique for that subinterface on the base interface.

The **list devices** command in the configuration process lists the interface identifiers. Here you can see the output of this command on a specific device:

```
Config>list devices
Interface          Connector     Type of interface
ethernet0/0        LAN1          Quicc Ethernet
serial0/0          SERIAL0/WAN1  AT COM
atm0/0             DSL1          ATM
bri0/0             BRI/ISDN1     ISDN Basic Rate Int
x25-node           ---           Router->Node
Config>
```

The first column indicates the name of the interface (*Interface*), the second column indicates the corresponding physical connector (*Connector*), and the third column specifies the type of interface programmed.

To select an interface you need to type its name, but you do not necessarily have to type all the characters of the name. For physical interfaces, simply type the beginning of the text in such a way that it does not match any other interface text, followed by the location if there is more than one interface of the same type. You don't always have to enter the interface location (X/X); this is only necessary when there are several interfaces of the same type (for example, several serial interfaces).

Examples:

```
Config>list devices
 Interface       Con    Type of interface            CSR     CSR2   int
 ethernet0/0     LAN1   Fast Ethernet interface   fa200e00           27
 serial0/0       WAN1   X25                       fa200a00  fa203c00   5e
 serial0/1       WAN2   X25                       fa200a20  fa203d00   5d
 serial0/2       WAN3   X25                       fa200a60  fa203f00   5b
 bri0/0          ISDN1  ISDN Basic Rate Int       fa200a40  fa203e00   5c
 x25-node        ---    Router->Node                     0             0
```

```
Config>
```

Here are some examples of valid commands that you can use to access the first serial interface on the motherboard (WAN 1) according to the devices listed in the previous table:

```
Config>network serial0/0
Config>network ser0/0
Config>network ser0
Config>network s0
```

Here are some examples of erroneous commands:

```
Config>network serial
Config>network ser
Config>network ser0/4
Config>network s7
```

The **network serial** command is incorrect because there are several serial interfaces on the device. Therefore, you need to specify the interface location.

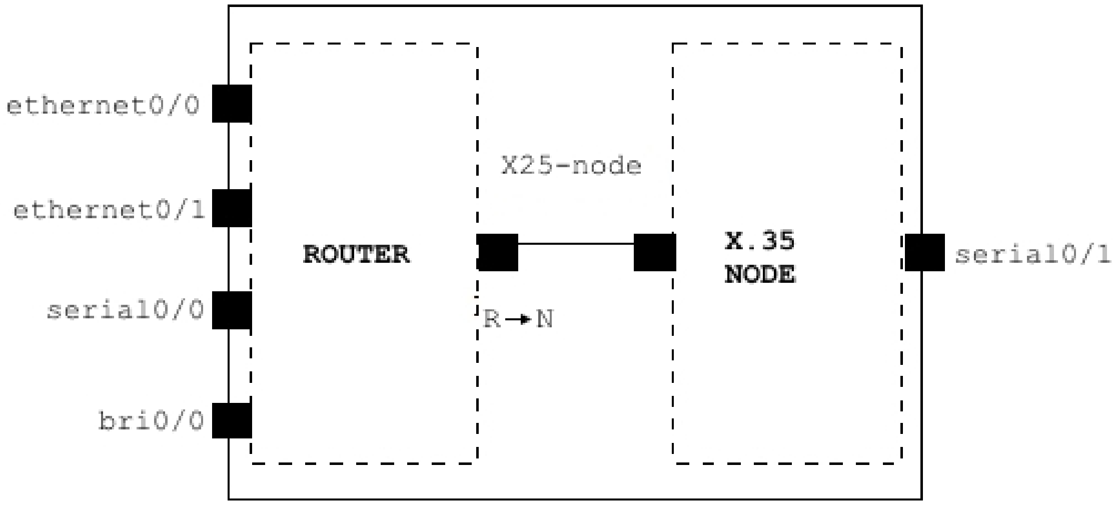
Here are some examples of valid commands that you can use to access the **bri** interface:

```
Config>network bri0/0
Config>network b0/0
Config>network bri0
Config>network b
```

In this case, as there is only one **bri** interface, you only need to type the text string of the interface and not the location. And you don't have to type the full text string. You can abbreviate it by typing just enough characters to distinguish the interface from all the others. In this case no other interface begins with the letter **b**, so one letter will suffice.

- It is important to note that some interfaces do not have an associated physical connector. This is the case of the x25-node interface example. This is because the interface itself allows the virtual machines to connect and therefore it does not have an external connector associated with it.

Using this information, you can redo the previous diagram for this case as follows:



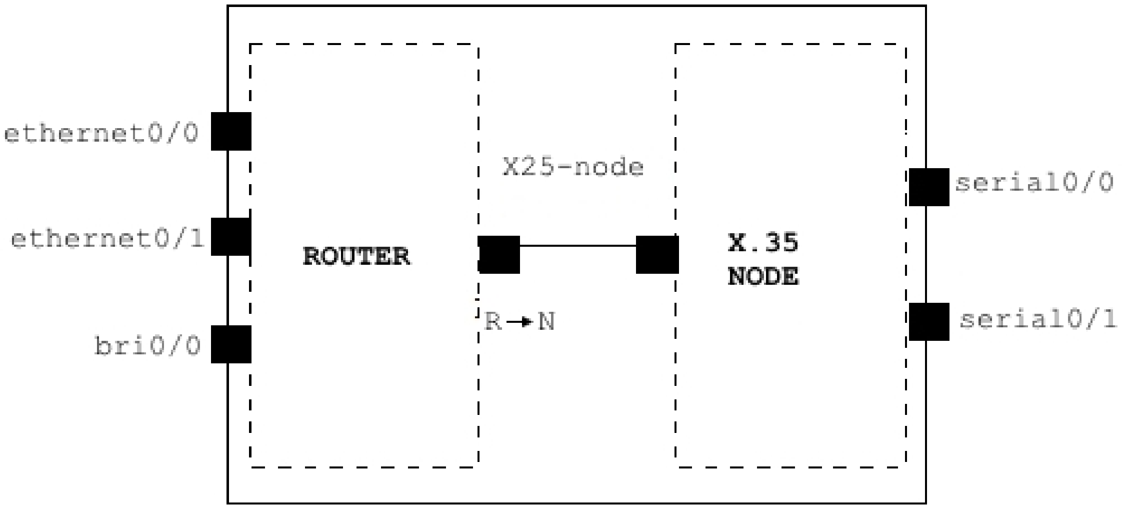
Now suppose you change the protocol of one of the WAN lines using the **set data-link** command and then you consult the interface table.

In the following example, the X25 protocol is assigned to the physical line 1:

```
Config>set data-link x25 serial0/0
Config>list devices

Interface          Connector     Type of interface
ethernet0/0        FE0/LAN1      Fast Ethernet interface
ethernet0/1        FE1/LAN2      Fast Ethernet interface
serial0/0          SERIAL0/WAN1  X25
serial0/1          SERIAL1/WAN2  X25
bri0/0             BRI/ISDN1     ISDN Basic Rate Int
x25-node           ---           Router->Node
Config>
```

As you can see, the router is now managing one more interface, while the node is managing one less.

The device scheme in this new example is:



The following example adds a generic Frame Relay interface over a basic ISDN access:

```
Config>add device fr 1
Config>
```

The interface identifier is a number between 1 and 9999 that allows you to distinguish the newly created interface from others of the same type, that is, from other frame-relay dial interfaces.

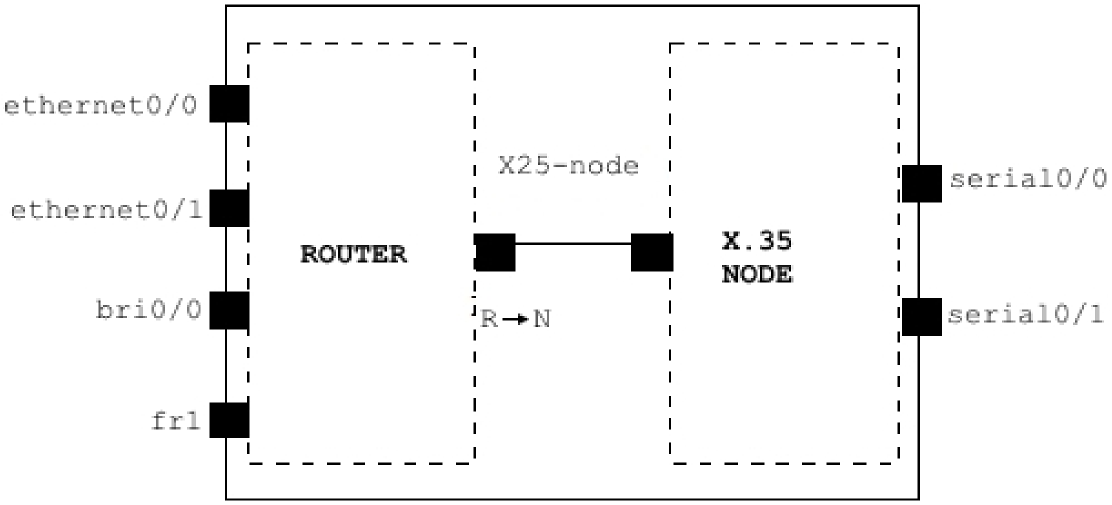
```
Config>list devices

Interface            Connector     Type of interface
ethernet0/0          FE0/LAN1      Fast Ethernet interface
ethernet0/1          FE1/LAN2      Fast Ethernet interface
serial0/0            SERIAL0/WAN1  X25
serial0/1            SERIAL1/WAN2  X25
bri0/0               BRI/ISDN1     ISDN Basic Rate Int
x25-node             ---           Router->Node
fr1                  ---           Generic FR
Config>
```

User-added dial interface names can be abbreviated by typing just enough characters to distinguish the interface from all the others. The identifier is mandatory.

Here are some examples of valid commands you can use to access the FR dial interface according to the devices listed in the previous table:

```
Config>network fr1
Config>network f1
```

The scheme of the resulting device in this new example is:



ATM interfaces can be configured with associated subinterfaces. For example, in a configuration with an xDSL card in SLOT 3:

```
Config>list devices
```

```
Interface            Connector     Type of interface
ethernet0/0          GE0/FE0/LAN1  GigabitEthernet interface
ethernet0/1          GE1/FE1/LAN2  GigabitEthernet interface
x25-node             ---           Router->Node
atm3/0               SLOT3         Generic ATM
Config>
```

We add a subinterface associated with this interface:

```
Config>add device atm-subinterface atm3/0 2
Config>list devices
Interface            Connector     Type of interface
ethernet0/0          GE0/FE0/LAN1  GigabitEthernet interface
ethernet0/1          GE1/FE1/LAN2  GigabitEthernet interface
x25-node             ---           Router->Node
atm3/0               SLOT3         Generic ATM
atm3/0.2             ---           ATM subinterface
Config>
```

To access this subinterface, you need to enter the base interface name followed by a period (.) and the subinterface ID. This ID must be different from that of any other subinterface on the same base interface (to be able to distinguish it from the others). You have to type it to access the subinterface even if there is only one. When naming the base interface, the same rules apply as for physical interfaces.

Here are some examples of valid commands you can use to access an **atm** subinterface according to the devices listed in the previous table:

```
Config>network atm3/0.2
Config>network atm3.2
Config>network atm.2
Config>network a.2
```

As there is only one ATM interface, you do not need to indicate its location.

2.2 Configuration process

The configuration process (**config** or **running-config**) allows you to configure router parameters such as:

- Interfaces
- Protocols

The CONFIG configuration process allows us to display and change the router's boot configuration and store it in flash memory or on a smart card. Any changes that you make in this process will not be stored unless you run the **save** command and will not take effect until you *restart* the router. To restart the router you can do two things:

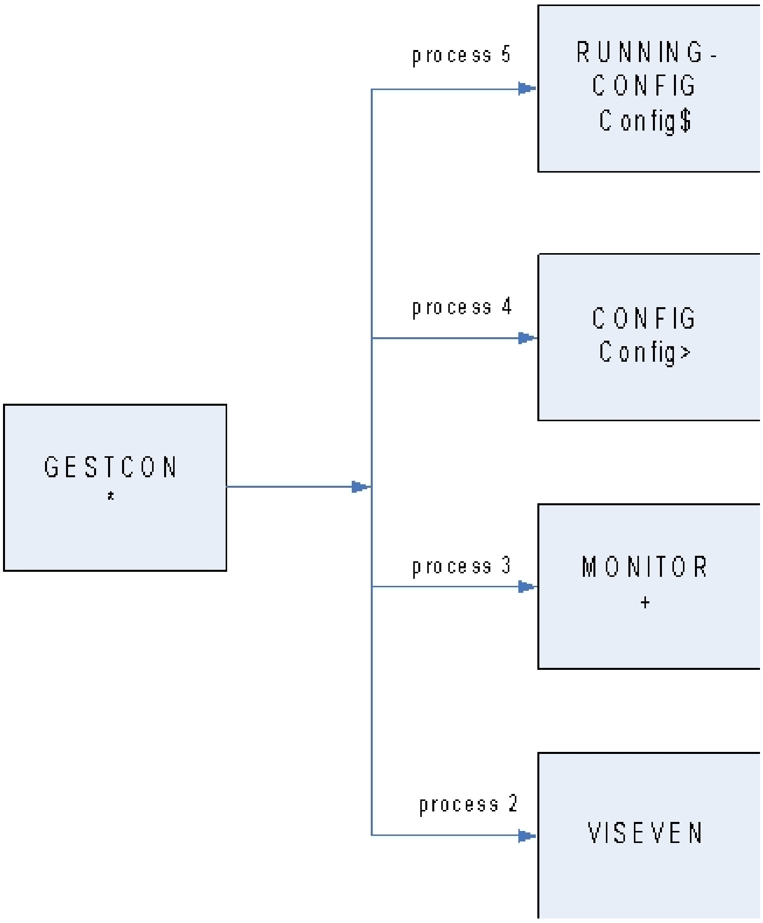
- Run the **restart** command from the Management Console prompt (***)**, or
- Turn the router off and on again.

The RUNNING-CONFIG configuration process allows us to dynamically change and display the router's active configuration and store it in flash memory or on a smart card. Any changes that you make in this process are not stored unless you run the **save** command. Changes made in the RUNNING-CONFIG process take effect immediately.

Note

The RUNNING-CONFIG process displays the active configuration but it does not allow you to modify all of it. For this reason, not all the CONFIG process commands (such as the *no config* command) are available in the RUNNING-CONFIG process.

The CONFIG and RUNNING-CONFIG processes fit into the router structure as follows:



2.3 Configuration process user interface

Entering and exiting the configuration process

To enter the CONFIG process from the GESTCON Management Console prompt (*), type the **config** command.

To enter the RUNNING-CONFIG process from the GESTCON Management Console prompt (*), type the **running-config** command.

Example:

```
*config
Config>
```

To exit the configuration process and return to the GESTCON Management Console prompt (*), type the escape character whose default value is (*Ctrl+p*).

Simultaneous access to configuration menus

When several users simultaneously access a device (via telnet or the console), the **Teldat Router** will block simultaneous access to certain configuration menus in order to avoid contradictions by parallel configuration processes.

If a conflict of this type does occur, the device prevents the user from accessing the configuration environment while informing him of the circumstance that led to the blocking.

Example:

```
+system telnet
  ID   USER        IP ADDRESS:PORT   CONNECTION TIME   INACTIVITY TIME
-------------------------------------------------------------------


  2   sample       192.168.1.2:1     08/03/05 12:29:26      0 min *
  1   root       172.24.51.128:131   08/03/05 12:28:59     29 min
+
```

- *“root” user console*

```
*config
Config>protocol ip
-- Internet protocol user configuration --
IP config>
```

- *“sample” user console*

```
*config
Config>protocol ip
CLI Error: Command locked by another user
CLI Error: Command error
Config>
```

In this case, the *sample* user intends to access the IP protocol configuration environment and is prevented from doing so by the device because the *root* user is accessing said menu at that moment.

Show command

The **show** command can be used in any menu in the configuration process. It allows three options:

```
Config>show ?
  all-config
  config
  menu
Config>
```

The **show all-config** command shows all menu and submenu configurations on the device that the user has access to. That is, it shows the console commands that would need to be entered to configure the device as it is in the moment the **show all-config** command is executed.

Example:

```
Config>show all-config
; Showing System Configuration for access-level 15 ...
; ATLAS50 Router 9 48 Version 10.7.0
log-command-errors
no configuration
;
network ethernet0/0
; -- Ethernet Interface User Configuration --
   ip address 172.24.78.116 255.255.0.0
;
;
;
;
;
exit
;
;
;
;
network ethernet0/1
; -- Ethernet Interface User Configuration --
   no ip address
;
exit
;
;
;
;
;
network x25-node
; -- X25-node interface configuration --
   no ip address
;
exit
;
protocol ip
; -- Internet protocol user configuration --
```

```
   classless
;
   tvrp
; -- TVRP Configuration --
      enable
;
      group 1 ip 172.24.78.128
      group 1 local-ip 172.24.78.116
;
   exit
;
exit
;
;
dump-command-errors
end
; --- end ---
Config>
```

This command shows all configurations on the router that the user has access to, regardless of the menu or submenu he is working in. For example from the IP menu:

```
IP config>show all-config
; Showing System Configuration for access-level 15 ...
; ATLAS50 Router 9 48 Version 10.7.0
log-command-errors
no configuration
;
network ethernet0/0
; -- Ethernet Interface User Configuration --
   ip address 172.24.78.116 255.255.0.0
;
;
;
;
;
exit
;
;
;
;
network ethernet0/1
; -- Ethernet Interface User Configuration --
   no ip address
;
exit
;
;
;
;
;
network x25-node
; -- X25-node interface configuration --
   no ip address
;
exit
;
protocol ip
; -- Internet protocol user configuration --
   classless
;
   tvrp
; -- TVRP Configuration --
      enable
;
      group 1 ip 172.24.78.128
      group 1 local-ip 172.24.78.116
;
```

```
   exit
;
exit
;
;
dump-command-errors
end
; --- end ---
IP config>
```

The **show config** command shows all menu and submenu configurations on the device that the user has access to and that are accessed through the menu the user is in when he runs the command.

Example:

```
Config>show config
; Showing System Configuration for access-level 15 ...
; ATLAS50 Router 9 48 Version 10.7.0
log-command-errors
no configuration
;
network ethernet0/0
; -- Ethernet Interface User Configuration --
   ip address 172.24.78.116 255.255.0.0
;
;
;
;
;
exit
;
;
;
;
network ethernet0/1
; -- Ethernet Interface User Configuration --
   no ip address
;
exit
;
;
;
;
;
network x25-node
; -- X25-node interface configuration --
   no ip address
;
exit
;
protocol ip
; -- Internet protocol user configuration --
   classless
;
   tvrp
; -- TVRP Configuration --
      enable
;
      group 1 ip 172.24.78.128
      group 1 local-ip 172.24.78.116
;
   exit
;
exit
;
;
dump-command-errors
end
```

```
; --- end ---
Config>
```

As you can see, running the **show config** command from the **root** menu is the same as running the **show all-config** command. However, if you run the **show config** command from the IP menu, only the IP and TVRP configurations are displayed:

```
IP config>show config
; Showing Menu and Submenus Configuration for access-level 15 ...
; ATLAS50 Router 9 48 Version 10.7.0
   classless
;
   tvrp
; -- TVRP Configuration --
      enable
;
      group 1 ip 172.24.78.128
      group 1 local-ip 172.24.78.116
;
   exit
;
IP config>
```

The **show menu** command shows the configuration of the menu from which the command is executed, but it does not show the configuration of its submenus.

Examples:

```
Config>show menu
; Showing Menu Configuration for access-level 15 ...
log-command-errors
no configuration
dump-command-errors
end
Config>
```

```
IP config>show menu
; Showing Menu Configuration for access-level 15 ...
   classless
;
IP config>
```

```
TVRP config>show menu
; Showing Menu Configuration for access-level 15 ...
      enable
;
      group 1 ip 172.24.78.128
      group 1 local-ip 172.24.78.116
;
TVRP config>
```

The configuration generated by the **show** command can be copied to a text file, edited at the user's convenience, and then pasted to an non-configured device in such a way that it gets configured.

If you want to abort the process while the router is showing the configuration, press the escape character (*Ctrl+p* by default).

Filtering the output of these commands is possible. To do this, we must add a pipe ("|") to the command, the filter we will use and the regular expression. If there is more than one word in the regular expression, please enter quotation marks. The filters allowed are:

- *begin*. Displays the result of the first occurrence of the regular expression.
- *end*. Displays the result until the first occurrence of the regular expression.
- *exclude*. Filters the **show** command output so that it excludes lines that contain the regular expression.
- *include*. Filters the **show** command output so that it only displays the lines that contain the regular expression.
- *more*. Displays of the result the number of lines specified. The default value is 25 lines.

```
*p 5

Config$show config | include "group"
```

```
      group 1 ip 172.24.78.128
      group 1 local-ip 172.24.78.116
```

Command history:

Release	Modification
11.01.05	The filtering options were introduced as of version 11.01.05.

Home command

The **home** command is available in all menus within the configuration process. It allows you to return to the configuration process prompt regardless of the current menu or submenu. Using this command, you don't have to leave the menus one by one with the "exit" command.

Examples:

```
*p 4


Config>protocol ip

-- Internet protocol user configuration --
IP config>ipsec

-- IPSec user configuration --
IPSec config>home


Config>
```

```
*p 5


Config$network ethernet0/0

-- Ethernet Interface User Configuration --
ethernet0/0 config$repeater-switch

-- Switch User Config --
ethernet0/0 switch config$home


Config$
```

Command history:

Release	Modification
11.01.03	This command was introduced as of version 11.01.03.

Root command

The **root** command is available in all menus that can be accessed during the configuration and monitoring processes. It allows you to return to the **root** prompt, regardless of the menu or submenu you are currently at.

Examples:

```
*
*p 4

Config>protocol ip

-- Internet protocol user configuration --
IP config>ipsec
```

```
-- IPSec user configuration --
IPSec config>root

*
```

Command history:

Release	Modification
11.01.06	This command was introduced as of version 11.01.06.

List of options

In many situations when you are configuring your device, you will need to choose an option to configure certain features of the router, such as Frame Relay PVCs (*Teldat Dm703-I Frame Relay*) or TVRP groups (*Teldat Dm725-I TVRP Protocol*).

Options can be abbreviated. You simply type just enough characters to distinguish an option from all the available ones.

Examples:

```
serial0/0 FR config>pvc 16 ?
  backup                Configure several backup parameters
  Bc                    Outgoing Committed Burst Size
  Be                    Outgoing Excess Burst Size
  CIR                   Outgoing Committed Information Rate
  compression           Enable/disable compression for this circuit
  default               Create the virtual circuit
  encapsulation         Encapsulation type
  encrypt               Enable/disable encryption
  fragmentation-size    Forced fragmentation size
  inverse-arp           Inverse ARP configuration for this dlci
  name                  Set the virtual circuit name
  no
  route                 Set static route for PVC switching
serial0/0 FR config>pvc 16 cir 32000
serial0/0 FR config>
```

In this case, you could also have typed **ci**, since no other option starts with **ci**. If you had typed the letter **c** and nothing else, you would have got an error message because there is another option starting with that letter (compression).

Transaction command

This command allows you to run transactions from the configuration menus. A transaction allows you to input and temporarily suspend configuration commands. At the end of the transaction, the commands that have been entered since the start of the transaction are executed and then applied to the configuration.

It is possible to abort an active transaction. In this case, the commands that have been entered since the beginning of the transaction are undone and are not applied to the configuration.

Optionally, you can configure the transaction so that no changes are applied in the configuration at the end of the transaction if any of the entered commands returns an error. This is useful if you are using a remote management application and you want to apply a set of configuration commands atomically.

You can run the **transaction** command from any configuration menu, either in the static configuration or in the dynamic configuration. There are four options:

```
Config$transaction ?
  abort            Terminate an active transaction discarding changes
  commit           Terminate an active transaction and apply changes
  fail-on-error    Make a transaction to fail if any command returns an error
  start            Start a CLI transaction
Config$
```

The **transaction abort** command rolls back an active transaction, which undoes any changes from commands entered since the start of the transaction.

The **transaction commit** command ends an active transaction, applying the changes from commands entered since the start of the transaction to the configuration. If you have configured the transaction not to apply changes when a command fails (using the **transaction fail-on-error** command) and some of the commands entered have generated an error, the detected errors are reported and no changes are applied.

The **transaction fail-on-error** command configures an active transaction to avoid applying changes upon termination of the transaction with the **transaction commit** command if any of the commands entered have returned an error.

The **transaction start** command initiates a transaction, at which time the entered commands start being registered to apply them together at the end of the transaction with **transaction commit**.

2.4 Configuration commands

In this section, we will describe the configuration commands (CONFIG and RUNNING CONFIG processes). Each command includes a description, syntax, and an example. The following table summarizes the configuration commands.

Command	**Function**
ADD	Creates a virtual interface.
AUTOINSTALL	Sets the auto-install parameters.
BACKUP-FILES	Backs up the system files.
BANNER	Configures banners on the device.
CONFIG-MEDIA	Specifies the active storage unit: flash, smart card or both.
CONFIRM-CFG-NEEDED	Confirms that the saved configuration is correctly configured.
CONFIRM-CFG	Confirms the current configuration.
COPY	Enables you to copy the settings from Running-Config to Config.
DESCRIPTION	Description of the configuration.
DISABLE	Disables a specific router feature.
DUMP-COMMAND-ERRORS	Displays command line errors.
ENABLE	Enables a specific router feature.
EVENT	Enters the event monitoring configuration process.
FEATURE	Defines additional router features, not associated with any default interface.
FILE	Allows you to perform file operations (**list** and **copy**).
FIRMWARE-CHECKING	Allows checking for firmware files when they are required.
FIXED-NUMBER-SNMP	Sets the order of the interfaces and classes.
FORMAT	Formats a storage unit on the device.
GLOBAL-PROFILES	Accesses the profile configuration menu for ATM, PPP, etc.
LICENCE-CHANGE	Changes the current licence.
LIST	Displays system parameters and hardware configuration.
LOG-COMMAND-ERRORS	Starts saving command line errors.
MANAGEMENT	Enters the master router configuration environment.
NETWORK	Enters the configuration menu for a specific interface.
NO	Undoes a command action or restores its default values.
NODE	Enters the X.25/ISDN Node, XOT or 270 configuration.
PRIVILEGE	Specifies custom execute permissions.
PROTOCOL	Enters a protocol's configuration.
RUSH-ENGINE	Rush engine configuration.
SAVE	Saves the configuration to the active storage unit.
SET	Configures system parameters, buffers, device name, etc.
STRONG-PASSWORD	Enables secure-password checking.
TELEPHONY	Sets Voice over IP parameters.
TIME	Allows you to view and change the system date and time.
UCI	Allows you to configure the **Teldat Router** encryption unit.
USER	Allows you to configure users.
END	End of configuration.

2.4.1 ADD

Allows you to create a virtual interface.

Syntax:

```
Config>add <option>
  device    Create a virtual device
```

- **< option >** specifies the selected option.

The only available option for this command is:

2.4.1.1 ADD DEVICE

Syntax:

```
Config>add device <virtual interface> [options]
```

- **<virtual interface>** is the type of virtual interface to create. To find out what types of interfaces are available, type **add device ?**:

```
Config>add device ?
  atm-subinterface     Create a virtual ATM Subinterface interface
  bvi                  Create a virtual Bridge interface
  bvi-subinterface     Create a virtual Bridge subinterface
  dial-routing         Create a virtual Dial-Route interface
  direct-ip            Create a virtual Direct IP interface
  eth-subinterface     Create a virtual Ethernet Subinterface interface
  fr                   Create a virtual Frame-Relay interface
  fr-subinterface      Create a virtual FR Subinterface interface
  hdlc                 Create a virtual HDLC interface
  l2tp                 Create a virtual L2TP interface
  loopback             Create a virtual Loopback interface
  ppp                  Create a virtual PPP interface
  rcellular            Create a virtual Remote Celullar interface
  tnip                 Create a virtual TNIP interface
  voip-cellular        Create a virtual VoIP interface over cellular network
  voip-dummy           Create a virtual VoIP interface with no hardware
  voip-isdn            Create a virtual VoIP interface over ISDN
  wlan-subinterface    Create a virtual Wireless LAN subinterface
  x25                  Create a virtual X25 interface
  xot                  Create a virtual XOT interface
```

- **[options]** The options that are available will depend on the type of virtual interface we want to create. They are described in the relevant interface manual.

Example:

```
Config>add device ppp ?
  <1..10030>    Interface Id
Config>add device ppp 1 ?
  <cr>
Config>add device ppp 1
Config>
```

For more information, please refer to the manual associated with the particular virtual interface you want to create.

2.4.2 AUTOINSTALL

Sets the device parameters to auto-install by frame relay from a TELDAGES network manager.

Syntax:

```
Config>autoinstall <parameter> [value]
  identifier         Configure identifier type
  management-host    Configure management host
```

- **< parameter>** is the identifier of the auto-install parameter to be configured.
- **[value]** is the value to be assigned to the parameter specified above.

Command history:

Release	Modification
11.00.03	This command is obsolete as of version 11.00.03.

There are two types of configurable parameters:

2.4.2.1 AUTOINSTALL IDENTIFIER

Specifies the type of device identifier for correct interpretation from the TELDAGES network manager.

Syntax:

```
Config>autoinstall identifier <type>
```

- **<type>** is the type of device identifier. Currently, only serial, which uses the device's serial number, is available to identify the device with the network manager.

Example:

```
Config>autoinstall identifier serial
Config>
```

2.4.2.2 AUTOINSTALL MANAGEMENT-HOST

Identifies the management station that will be used to perform remote autoconfiguration.

Syntax:

```
Config>autoinstall management-host <host identifier>
```

- **<host identifier>** is the management station identifier. This identifier can be an IP address or a *Fully Qualified Domain Name/Hostname.*

Example:

```
Config>autoinstall management-host www.gestion.teldat.es
Config>
```

2.4.3 BACKUP-FILES

Creates a recovery point by making a copy of the system files from the primary partition to the backup partition. In this way, if the file system were to be corrupted, the device would automatically restore files from the recovery point and remain functional.

The **configuration monitoring** command displays information about the recovery process, indicating whether the device booted normally (from the primary partition) or whether it restored from a recovery point (from the backup partition).

The **[no] firmware-checking** configuration command determines whether file system restore is enabled in case any required firmware files are corrupted.

Note

This command only works on devices with partitioned flash memory.

Syntax:

```
Config>backup-files
```

Example 1:

```
Config>backup-files
Backup in progress...
Backup successful.
Config>
```

In this example, a recovery point has been created successfully.

Example 2:

```
Config>backup-files
CLI Error: Backup device not available.
CLI Error: Command error
Config>
```

In this example the device did not have a partitioned flash, so the command returned an error and no recovery point was created.

2.4.4 BANNER

Allows you to configure a banner on the device.

Syntax:

```
Config>banner <type>
  login    Set login banner
  exec     Set exec banner
```

< type > specifies the type of banner to configure.

2.4.4.1 Banner login

Use this command to configure an access banner to be displayed whenever a user tries to connect to the device through the console, telnet or FTP.

If the banner contains more than one line of text, each line must be configured separately using the **banner login** command. The lines of text should be entered in the order you want to display them. If a line of text contains spaces, it must be enclosed in quotation marks.

The size of the banner is limited to a maximum of 15 lines of text and 80 characters per line.

Syntax:

```
Config>banner login <line of text>
```

Example:

```
Config>banner login "###########################################################################"
Config>banner login "#Este equipo es propiedad de Sample, S.A. y su uso está restringido a sus #"
Config>banner login "#  empleados. Por favor, aborte esta conexión si usted no es empleado de  #"
Config>banner login "#  Sample, S.A. o tiene una autorización legal para acceder al equipo.    #"
Config>banner login "###########################################################################"
Config>show config
; Showing System Configuration for access-level 15 ...
; ATLAS50 Router 9 48 Version 10.7.0
log-command-errors
no configuration
banner login "################################################################################"
banner login "#  Este equipo es propiedad de Sample, S.A. y su uso está restringido a sus    #"
banner login "#  empleados. Por favor, aborte esta conexión si usted no es empleado de       #"
banner login "#  Sample, S.A. o tiene una autorización legal para acceder al equipo.      #"
banner login "################################################################################"
;
;
;
;
network ethernet0/0
; -- Ethernet Interface User Configuration --
   no ip address
;
exit
;
;
;
network ethernet0/1
; -- Ethernet Interface User Configuration --
   no ip address
;
exit
;
```

```
;
;
;
network x25-node
; -- X25-node interface configuration --
   no ip address
;
exit
;
;
;
dump-command-errors
end
; --- end ---
Config>save yes

Building configuration as text... OK
Writing configuration... OK on Flash
Config>
*restart
Are you sure to restart the system(Yes/No)? y
  Done
Restarting. Please wait ..................................................
APP DATA DUMP......................................................................
 Bios-stack used: 0x1498
 Bios-stack free: 0x2B68
 Aux-stack used: 0x0
 Aux-stack free: 0x2000
Running application
Flash configuration read
Parsing text mode configuration ...
Configuration parsed
Initializing
Press any key to get started

################################################################################
#  Este equipo es propiedad de Sample, S.A. y su uso está restringido a sus    #
#  empleados. Por favor, aborte esta conexión si usted no es empleado de       #
#  Sample, S.A. o tiene una autorización legal para acceder al equipo.         #
################################################################################
User:
```

By default, there is no access banner configured on the device.

2.4.4.2 Banner exec

Use this command to configure a welcome banner to be displayed when a user is granted access to the device through the console, Telnet, SSH or FTP.

If the banner contains more than one line, each line must be configured separately using the **banner exec** command. The lines should be entered in the order you want to display them. If a line of text contains spaces, it must be enclosed in quotation marks.

The banner is limited to a maximum of 15 lines of text and 80 characters per line.

Syntax:

```
Config>banner exec <text>
```

Example:

```
Config>banner exec "##############################################################"
Config>banner exec " "
Config>banner exec "............:::::::::::    WELCOME     ::::::::::............."
Config>banner exec " "
Config>banner exec "##############################################################"
Config>save yes
Building configuration as text... OK
Writing configuration... OK on Flash as CONFIG
```

```
Config>
*restart
Are you sure to restart the system(Yes/No)? y
Restarting. Please wait ..............................................
APP DATA DUMP...................................................
 Bios-stack used: 0x14E8
 Bios-stack free: 0x2B18
 Aux-stack used: 0x0
 Aux-stack free: 0x2000
Running application
WARNING: Running with super license.
Flash configuration read
Parsing text mode configuration ...
Configuration parsed
Initializing
Press any key to get started
Teldat               (c)2001-2012

Router model Super * * CPU MPC8260  S/N: 472/04709
2 LAN, 2 WAN Lines, 1 ISDN Line
CIT software version: 11.00.00 Oct 25 2012 18:43:58

##############################################################
...........::::::::::    WELCOME     ::::::::::...........

##############################################################
*
```

2.4.5 CONFIG-MEDIA

Use this command to select the active storage device. The configuration will be saved to this device when you apply the **save** configuration command. It is also the default media in the **file list** command.

The devices can use two types of storage media: flash memory and smart card. All devices use flash as their primary storage media. Not all device models support smart cards. Please refer to your device's installation manual to find out whether your device supports smart card technology.

By default, devices that support smart cards are configured with the *smartcard-flash* value while those that do not are configured with the *flash* option. You can check which drive is active by using the **file list** configuration command.

Syntax:

```
Config>config-media <media>
  flash                Configurations are stored only in Flash
  smartcard            Configurations are stored only in SmartCard
  smartcard-flash      Configurations are stored in SmartCard and Flash
```

- **< media >** specifies the name of the storage drive to be used.

For more information on using storage units, refer to the **file list** and **file copy** commands in this manual and your device's installation manual. For more information on how to select the active configuration file, refer to your device's installation manual.

The behavior described below is slightly different when the **e** microswitch is set to ON. For more information, please see your device's installation manual.

2.4.5.1 CONFIG-MEDIA FLASH

Specifies that flash memory is to be used as the only active storage device when reading or writing configurations.

The device reads the configuration from flash when powering up. If it can't find the active configuration file, it boots from the default configuration instead. The **save** command saves the configuration to flash under the active filename.

Example:

```
Config>config-media flash
```

2.4.5.2 CONFIG-MEDIA SMARTCARD

Specifies that the smart card is to be used as the only active storage device when reading or writing configurations.

The device reads the configuration from the *smart card* when powering up. If it cannot detect the smart card or find the active configuration, it boots from the default configuration instead. The **save** command saves the configuration to the *smart card* under the active filename.

Example:

```
Config>config-media smartcard
```

2.4.5.3 CONFIG-MEDIA SMARTCARD-FLASH

Specifies that both the smart card and flash memory are to be used as the active storage devices when reading or writing configurations. The smart card takes priority over flash.

The device reads the configuration from the *smart card* when powering up. If it cannot detect the smart card or find the active configuration, it repeats the operation in flash. If it cannot find the file in flash, it boots from the default configuration instead. If, after reading the configuration from the smart card, the device ascertains that the file is not in flash, it writes it in to synchronize the two media.

The **save** command *saves the configuration to both the smart card and flash* under the active filename. The devices used to store the configuration show up on the console, as does a warning text if the recording was not possible on either of the devices.

Example:

```
Config>config-media smartcard-flash
```

2.4.6 CONFIRM-CFG

Use this command to confirm that the current configuration is configured correctly. You must have saved a configuration with the confirm saved configuration function enabled (see the **confirm-cfg-needed** command).

Syntax:

```
Config>confirm-cfg
```

You can also perform this action via SNMP. See *CONFIRM-CFG-NEEDED* on page 36.

2.4.7 CONFIRM-CFG-NEEDED

Use this command to confirm that the saved configuration is configured correctly. When you run this command, the configuration is saved in **TEMP.CFG** for testing. If, after restarting the device, you don't save the configuration with the **confirm-cfg** command within the programmed time, the device boots with the previous configuration. If you confirm the configuration, it is saved under its corresponding name; so the use of **TEMP.CFG** is transparent to the user. If the new configuration causes the device to restart before reaching the programmed test time, the previous configuration will be restored after ten restarts without the configuration being confirmed. If you use the **no confirm-cfg** command, the device restarts with the previous configuration. The **no confirm-cfg-needed** command disables the requirement that new saved configurations be confirmed.

Syntax:

```
Config>confirm-cfg-needed <option>
  default     Enables the need of configuration confirmation
  timeout     Sets a timeout to wait for confirmation
```

- **< option >** specifies the selected action.

2.4.7.1 CONFIRM-CFG-NEEDED DEFAULT

Enables the requirement that the saved configurations be confirmed within a 10-minute time period.

Syntax:

```
Config>confirm-cfg-needed default
```

Example:

```
Config>confirm-cfg-needed default
Config>
```

2.4.7.2 CONFIRM-CFG-NEEDED TIMEOUT

Configures the test period for the new configuration. This is the amount of time the device will wait before restarting the previous configuration if the new configuration is not confirmed. The minimum value is one minute and the maximum is 5 weeks.

Syntax:

```
Config>confirm-cfg-needed timeout <time>
```

- **< option >** specifies the test period in any of the following formats: Xw, Xd, Xh, Xm, Xs, HH:MM, HH:MM:SS.

Example:

```
Config>confirm-cfg-needed timeout 30s
Config>
```

It is possible to run the confirm saved configuration feature via SNMP. For this purpose, three new SNMP variables have been created with their corresponding OIDs:

- **telAdminStatusConfirmConfig** (OID: 1.3.6.1.4.1.2007.1.2.14). This variable is used to confirm/not confirm the current configuration.

  Setting this variable to 1 in a write operation confirms the current configuration. Setting it to 0 rejects the current test configuration so that the device restarts with the previous configuration.

  A 0 value (confirmed) in a read operation indicates that the current configuration has been confirmed. A value of 1 (test_cnfg_inactive) indicates that the current configuration has not yet been confirmed (therefore it is a test configuration) and is not active; the previous configuration is the active configuration. A value of 2 (test_cnfg_active) indicates that the current configuration has not yet been confirmed (i.e., it is a test configuration) but it is the active configuration. A value of 3 (configuration_recovered) indicates that the previous configuration has been recovered, confirmed and is active. A value of 4 (undefined) is only given if the confirm saved configuration feature is disabled because, in this case, the use of this SMNP variable is meaningless.

  There are three scenarios in which the old configuration can be recovered (and therefore the telAdminStatusConfirmConfig variable would get the value 3 of configuration_recovered):

  a) After the confirmation timer expires without confirming the test configuration.

  b) After 10 consecutive device restarts (produced before the confirmation timer expires). Restoring the old configuration in this scenario is meant to protect us in the event the new configuration forces the device to restart before reaching the scheduled test time.

  c) After rejecting the current test configuration with the **no confirm-cnfg** command (or via SNMP by typing a **0** in the telAdminStatusConfirm Config variable).

- **telAdminStatusConfirmEnabled** (OID: 1.3.6.1.4.1.2007.1.2.15). This variable is used to enable/disable the confirm saved configuration feature.

  Setting this variable to 1 in a write operation enables the need to confirm the saved configuration. Setting it to 0, disables the need to confirm new saved configurations.

  A 0 value (disable) in a read operation indicates that the confirm configuration feature is disabled, while a value of 1 (enable) indicates that the confirm configuration feature is enabled.

- **telAdminStatusTimeoutConfirm** (OID: 1.3.6.1.4.1.2007.1.2.16). This variable is used to configure the value of the confirm saved configuration feature timer (in seconds). That is, you use it to set the amount of time the device waits before restarting the previous configuration if the current configuration is not confirmed.

  Valid values for this variable range from 60 (1 minute) to 3024000 (5 weeks). Its default value is 600 (10 minutes).

  In a write operation, values between 60 and 3024000 set the timer to the input value.

  In a read operation, values between 60 and 3024000 return the seconds to the timer value.

Therefore, a logical operating sequence for the confirm saved configuration feature via SNMP is:

- Initially, the confirm saved configuration feature is disabled. The SNMP variables have the following values:

  - telAdminStatusConfirmConfig = 4 (undefined)

  - telAdminStatusConfirmEnabled = 0 (disable)

  - telAdminStatusTimeoutConfirm = 600

- We enable the confirm saved configuration feature (setting the telAdminStatusConfirmEnabled SNMP variable value to 1). The SNMP variables have the following values:

  - telAdminStatusConfirmConfig = 0 (confirmed)

  - telAdminStatusConfirmEnabled = 1 (enable)

  - telAdminStatusTimeoutConfirm = 600

- We set the confirmation timer value to the desired value (for example, 1 minute). To do this, set the telAdminStatusTimeoutConfirm SNMP variable value to 60. The SNMP variables have the following values:

  - telAdminStatusConfirmConfig = 0 (confirmed)

  - telAdminStatusConfirmEnabled = 1 (enable)

  - telAdminStatusTimeoutConfirm = 60

- We make the appropriate changes to the device configuration and save them (setting the telAdminStatusSaveConfig SNMP variable value to 1). The SNMP variables have the following values:

  - telAdminStatusConfirmConfig = 1 (test_cnfg_inactive)

  - telAdminStatusConfirmEnabled = 1 (enable)

  - telAdminStatusTimeoutConfirm = 60

- We restart the device. The SNMP variables have the following values:

  - telAdminStatusConfirmConfig = 2 (test_cnfg_active)

  - telAdminStatusConfirmEnabled = 1 (enable)

  - telAdminStatusTimeoutConfirm = 60

- We confirm the (active) test configuration before the 60 second timer expires (setting the telAdminStatusConfirmConfig SNMP variable value to 1). The SNMP variables have the following values:

  - telAdminStatusConfirmConfig = 0 (confirmed)

  - telAdminStatusConfirmEnabled = 1 (enable)

  - telAdminStatusTimeoutConfirm = 60

- Continuing the example, we could make further changes to the device configuration and save them (by setting the telAdminStatusSaveConfig SNMP variable to 1). The SNMP variables have the following values:

  - telAdminStatusConfirmConfig = 1 (test_cnfg_inactive)

  - telAdminStatusConfirmEnabled = 1 (enable)

  - telAdminStatusTimeoutConfirm = 60

- We restart the device, noting the test configuration is active. The SNMP variables show the following values:

  - telAdminStatusConfirmConfig = 2 (test_cnfg_active)

  - telAdminStatusConfirmEnabled = 1 (enable)

  - telAdminStatusTimeoutConfirm = 60

- Now we let the timer expire (1 minute) without confirming the test configuration. Or we can restore the previous configuration by setting the telAdminStatusConfirmEnabled SNMP variable value to 0. In both cases, the device restarts with the previous configuration as the active configuration, and the SNMP variables show the following values:

  - telAdminStatusConfirmConfig = 3 (configuration_recovered)

  - telAdminStatusConfirmEnabled = 1 (enable)

  - telAdminStatusTimeoutConfirm = 60

2.4.8 COPY

Allows you to copy the configuration from running-config (P5) to config (P4). The device asks for confirmation if the config configuration (P4) has been modified.

Syntax:

```
Config>copy ?
  config-file       Copy a config. file to Config.
  running-config    Copies Running Config
```

2.4.8.1 COPY CONFIG-FILE

Command history:

Release	Modification
11.00.04	This command option was introduced as of version 11.00.04.

Allows you to apply a configuration file (previously saved to flash) to config (P4).

Syntax:

```
Config>copy config-file <file-name> config
```

Example:

```
Config>copy config-file prueba.cfg config
Configuration copied OK
Config>
```

2.4.8.2 COPY RUNNING-CONFIG

Allows you to apply the running-config file (P5) to config (P4).

Syntax:

```
Config>copy running-config config [yes]
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.

Example:

```
Config>copy running-config config
Warning: Static-config has been changed.
Copy Running-Config to Config(Yes/No)? y
Copying configuration... OK
Config>
```

Command history:

Release	Modification
11.01.06	The "[yes]" option was introduced as of version 11.01.06.

2.4.9 DESCRIPTION

Adds descriptive texts about the device configuration. This description is displayed on the screen when you run the **list configuration** command.

Syntax:

```
Config>description <text>
```

- **< text >** is the configuration description text. If the text contains spaces, it must be enclosed in quotation marks (e.g., description “description with spaces”).

Example:

```
Config>description Madrid_Branch_Outcoming_Router
```

2.4.10 DISABLE

Disables a specific customizable parameter. This command disables the behavior enabled by the **enable patch <parameter>**. In order to use it, you need to know the name of the parameters enabled. To find out this information, use the **list patch** command in the configuration console.

Syntax:

```
Config>disable patch <id>
```

- **< id >** is the name of the parameter to disable. If we enter the **default** parameter name, ALL the active parameters will be disabled.

Example:

```
Config>disable patch arpi_snd_lcl
Config>
```

2.4.11 DUMP-COMMAND-ERRORS

Displays a list of the first five erroneous commands entered in the configuration console since the **log-command-errors** command was used. This command is especially useful for detecting errors that occurred when loading an entire configuration file to a device.

Syntax:

```
Config>dump-command-errors
```

Example:

```
Config>dump-command-errors
Warning: possible errors in the configuration, at least these found:
line 2 -> ast
line 4 -> dev eth1
line 5 -> dev ser134
line 6 -> conf 0
line 7 -> list interf eth1
(lines counting since last log-command-errors command)
Too many errors, some cannot be printed (printed 5 of 8)
Config>
```

2.4.12 ENABLE

Enables a specific customizable parameter. This command is used to modify the behavior of the router *in certain circumstances*. It is used when you need to manage custom versions. In order to use it, you need to know the name of the available parameters (the customizable parameters relating to each router functionality are duly documented in the manuals associated with those functionalities) and the possible values that they support. To enable a parameter, you need to enter its name and the desired value.

Syntax:

```
Config>enable patch <id> [value]
```

- **< id >** is the name of the parameter to be enabled.
- **[value]** is the value of the parameter.

Example:

```
Config>enable patch arpi_snd_lcl 1
Config>
```

To check which parameters are currently active on your device, use the **list patch** command. To disable an active customizable parameter, use the **disable patch** command followed by the parameter name.

2.4.13 EVENT

Records events stored by the Event Logging System as configuration items. Type **exit** to return to the Config> prompt.

Syntax:

```
Config>event
```

Example:

```
Config>event
-- ELS Config --
ELS Config>
```

To find out which commands are available from this prompt, please see *Event Logging System ELS* on page 154.

2.4.14 FEATURE

Defines additional router features, not associated with any default interface.

Syntax:

```
Config>feature <option> [parameters]
  aaa                        AAA configuration environment
  access-lists               Access generic access lists configuration
                             environment
  act                        Alsa custom trap configuration environment
  afs                        Advanced stateful firewall and routing
  autoset-cfg                Autoset-Config configuration environment
  bandwidth-reservation      Bandwidth-Reservation configuration environment
  class-map                  Class Map configuration environment
  control-access             Control-access configuration environment
  cwmp                       CPE Wan Management Protocol
  dns                        DNS configuration environment
  dns-updater                DNS Updater configuration environment
  echo-responder             Echo protocol configuration environment
  err-disable                Error disable configuration
  external-dying-gasp        External dying gasp configuration
  external-tcms              External TCMS configuration
  frame-relay-switch         Frame RRelay Switch configuration environment
  gps-applications           GPS applications configuration environment
  hotspot                    Hotspot configuration environment
  http                       Access the router http protocol configuration
  ip-discovery               TIDP configuration environment
  istud                      IPSEC Tunnel Server Discovery configuration environment
  key-chain                  Key chain management
  ldap                       LDAP configuration environment
  mac-filtering              Mac-filtering configuration environment
  management                 Management configuration environment
  netflow                    Netflow client configuration
  nsla                       Network Service Level Advisor configuration
  nsm                        Network Service Monitor configuration environment
  ntp                        NTP configuration environment
  policy-map                 Policy Map configuration environment
  power-switch               TeleControl Module control environment
  prefix-lists               Access generic prefix lists configuration
                             environment
  radius                     RADIUS protocol configuration environment
  rmon                       Remote Network Monitoring configuration environment
  route-map                  Route-map configuration environment
  scada-forwarder            SCADA Forwarder configuration environment
  sniffer                    Sniffer configuration environment
  spi                        SPI, mobile IP Presence Service,configuration
                             environment
  ssh                        Secure Shell configuration environment
  stun                       Stun facility configuration environment
  syslog                     Syslog configuration environment
  tftp                       TFTP configuration environment
  tms                        TMS configuration environment
  vlan                       IEEE 802.1Q switch configuration environment
  vli                        Virtual Linux Interface configuration
  vrf                        VRF configuration environment
  wnms                       Wireless Network Management System
  wrr-backup-wan             WRR configuration environment
  wrs-backup-wan            WRS configuration environment
```

- **< option >** is the name of the feature to be configured.
- **[parameters]** are the parameters required for the specified option.

2.4.14.1 FEATURE AAA

Accesses the *AAA* feature configuration environment.

Syntax:

```
Config>feature aaa
```

Example:

```
Config>feature aaa
-- AAA user configuration --
AAA config>
```

For further information on how to set up the AAA feature, please see the following manual: *Teldat Dm800-I AAA Feature*.

2.4.14.2 FEATURE ACCESS-LISTS

Accesses the generic *access list* configuration environment.

Syntax:

```
Config>feature access-lists
```

Example:

```
Config>feature access-lists
-- Access Lists user configuration --
Access Lists config>
```

For further information on how to configure generic access lists, please see the following manual: *Teldat Dm752-I Access Control*.

2.4.14.3 FEATURE ACT

Accesses the *ACT* feature's configuration menu.

Syntax:

```
Config>feature act
```

Example:

```
Config>feature act
-- Alsa Custom Trap configuration --
ACT config>
```

For further information on how to configure the ACT feature, please see the following manual: *Teldat Dm818-I ACT Feature*.

2.4.14.4 FEATURE AFS

Accesses the *afs* configuration environment.

Syntax:

```
Config>feature afs
```

Example:

```
Config>feature afs
AFS config>
```

For further information on how to configure *afs*, please see the following manual: *Teldat Dm786-I AFS* .

2.4.14.5 FEATURE AUTOSET-CFG

Allows routers that support *autoset-cfg* to access the *autoset-cfg* configuration environment.

Syntax:

```
Config>feature autoset-cfg
```

Example:

```
Config>feature autoset-cfg
-- Autosetcfg Configuration --Autoset-cfg Config>?
```

For further information on how to configure *autoset-cfg*, please see the following manual: *Teldat Dm797-I Dynamic Configuration Control*.

2.4.14.6 FEATURE BANDWIDTH-RESERVATION

Accesses the *Bandwidth-Reservation* configuration environment (BRS).

Syntax:

```
Config>feature bandwidth-reservation
```

Example:

```
Config>feature bandwidth-reservation
-- Bandwidth Reservation user configuration --
BRS Config>
```

For further information on how to configure bandwidth reservation, please see the following manual: *Teldat Dm715-I Bandwidth Reservation System.*

2.4.14.7 FEATURE CLASS-MAP

Accesses the *class-map* configuration environment.

Syntax:

```
Config>feature class-map
```

Example:

```
Config>feature class-map
-- Class-Map Menu Configuration --
Class-map Config>
```

For further information on how to configure *class-map*, please see the following manual: *Teldat Dm795-I Policy Map Class Map*.

2.4.14.8 FEATURE CONTROL-ACCESS

Accesses the *control-access* configuration environment for the Corporate Encryption System. This environment is only accessible on devices with encryption cards.

Syntax:

```
Config>feature control-access
```

Example:

```
Config>feature control-access
CtrlAcc Config>
```

2.4.14.9 FEATURE CWMP

Accesses the *CPE Wan Management Protocol* configuration environment. For more information, please see the following manual: *Teldat Dm826-I CPE Wan Management Protocol (CWMP)* .

Syntax:

```
Config>feature cwmp
```

Example:

```
Config>feature control-access
-- CPE WAN Management Protocol configuration --
CWMP Config>
```

Command history:

Release	**Modification**
11.01.02	The "*cwmp*" feature was introduced as of version 11.01.02.

2.4.14.10 FEATURE DNS

Accesses the *DNS* configuration environment. For more information, please see the following manual: *Teldat Dm723-I DNS Client*.

Syntax:

```
Config>feature dns
```

Example:

```
Config>feature dns
-- DNS resolver user configuration --
DNS config>
```

2.4.14.11 FEATURE DNS-UPDATER

Accesses the *dns-updater* configuration environment.

Syntax:

```
Config>feature dns-updater
```

Example:

```
Config>feature dns-updater
-- DNS UPDATER configuration --
DNS UPDATER config>
```

For further information on how to configure *dns-updater*, please see the following manual: *Teldat Dm785-I DNS Updater*.

2.4.14.12 FEATURE ECHO-RESPONDER

Accesses the *echo-responder* configuration environment.

Syntax:

```
Config>feature echo-responder
```

Example:

```
Config>feature echo-responder
-- ECHO user configuration --
ECHO config>
```

For further information on how to configure the *echo-responder*, please see the following manual: *Teldat Dm702-I TCP IP*.

2.4.14.13 FEATURE ERR-DISABLE

Accesses the *err-disable* configuration environment.

Syntax:

```
Config>feature err-disable
```

Example:

```
Config>feature err-disable
-- Error Disable user configuration --
errdisable config>
```

2.4.14.14 FEATURE EXTERNAL-DYING-GASP

Accesses the external-dying-gasp configuration environment.

This environment can only be accessed by devices that support DGe .

Syntax:

```
Config>feature external-dying-gasp
```

Example:

```
Config>feature external-dying-gasp
-- Frame Relay Switch configuration --
Ext-dying-gasp Switch>
```

The available commands at the feature's menu include:

```
Ext-dying-gasp Config> ?
  enable    Enable external dying gasp feature
  no        Negate a command or set its defaults
  exit      Exit to parent menu
```

The ext-dying-gasp configuration enables or disables DGe signal detection.

Command history:

Release	Modification
11.01.08	The "*feature external dying-gasp*" command was introduced as of version 11.01.08

2.4.14.15 FEATURE EXTERNAL-TCMS

Accesses the external-tcms configuration environment.

Syntax:

```
Config>feature external-tcms
```

Example:

```
Config>feature external-tcms
-- External tcms Configuration --
Ext-tcms Config>
```

The available commands at the feature's menu include:

```
Ext-tcms Config> ?
  enable    Enable external tcms feature
  no        Negate a command or set its defaults
  exit      Exit to parent menu
```

```
Ext-tcms Config> enable ?
  active-line      Line is active
  inactive-line    Line is inactive
```

When enabled, the ext-tcms configuration will provide two options: activate the line or set an inactive line. The external-tcms option is not enabled by default.

Please note that, when *feature external-tcms* is enabled, the console will no longer appear.

Operation scripts can also be added to this command.

Example:

```
MGT config> show config

      operation 1 system script 1
      operation 1 track event ETH.050
;
      operation 2 system script 2
      operation 2 track event ETH.051
;
      script 1 commands "p 5"
      script 1 commands "feature external-tcms"
      script 1 commands "enable active-line"
;
      script 2 commands "p 5"
      script 2 commands "feature external-tcms"
      script 2 commands "enable inactive-line"
;
```

This configuration activates the line when a working ethernet is detected and disables the line configuration when an ethernet is identified as down. For more information, please see the scripts section.

Command history:

Release	**Modification**
11.01.08	The "*feature external-tcms*" command was introduced as of version 11.01.08

2.4.14.16 FEATURE FRAME-RELAY-SWITCH

Accesses the Frame Relay switch configuration environment.

For more information, please see the following manual: *Teldat Dm703-I Frame Relay* .

Syntax:

```
Config>feature frame-relay-switch
```

Example:

```
Config>feature frame-relay-switch
-- Frame Relay Switch configuration --
Frame Relay Switch>
```

2.4.14.17 FEATURE GPS-APPLICATIONS

Allows devices that support *gps-applications* to access the *gps-applications* configuration environment.

Syntax:

```
Config>feature gps-applications
```

Example:

```
Config>feature gps-applications
-- GPS Applications Configuration --
GPS-Apps Cfg>
```

For further information on how to configure *gps-applications*, please see the following manual: *Teldat Dm812-I GPS* .

2.4.14.18 FEATURE HOTSPOT

Accesses the *HotSpot* feature's configuration menu. For more information, please see the following manual: *Dm820-I HotSpot Feature*.

Syntax:

```
Config>feature hotspot
```

Example:

```
Config>feature hotspot

-- Hotspot configuration --
HS config>
```

Command history:

Release	**Modification**
11.00.03	The "*Hotspot*" feauture was introduced as of version 11.00.03.

2.4.14.19 FEATURE HTTP

Accesses the router's *http* protocol configuration. This command (or functionality) is not available on all models.

Syntax:

```
Config>feature http
```

Example:

```
Config>feature http
```

```
-- HTTP user configuration --
HTTP config>
```

2.4.14.20 FEATURE IP-DISCOVERY

Accesses the *TIDP* (Teldat IP Discovery Protocol) configuration environment.

Syntax:

```
Config>feature ip-discovery
```

Example:

```
Config>feature ip-discovery
-- Teldat IP Discovery Protocol configuration --
TIDP config>
```

2.4.14.21 FEATURE IPV6-ACCESS-LIST

Accesses the *ipv6-access-list* configuration environment.

Syntax:

```
Config>feature ipv6-access-list
```

Example:

```
Config>feature ipv6-access-list
-- IPv6 Access Lists user configuration --
IPV6 Access Lists config>
```

For further information on how to configure IPv6 access lists, please see the following manual: *Teldat Dm808-I IPv6 Access Control*.

2.4.14.22 FEATURE ISTUD

Allows devices that support *istud* to access the *istud* configuration environment.

Syntax:

```
Config>feature istud
```

Example:

```
Config>feature istud
-- ISTUD configuration --
ISTUD config>
```

For further information on how to configure *istud*, please see the following manual: *Teldat Dm784-I ISTUD Feature* .

2.4.14.23 FEATURE KEY-CHAIN

Accesses the *key-chain* configuration environment.

Syntax:

```
Config>feature key-chain
```

Example:

```
Config>feature key-chain
-- Key Chain user configuration --
Key-chain Config>
```

For further information on how to configure *key-chain* , please see the following manual: *Teldat Dm792-I Key Management*.

2.4.14.24 FEATURE LDAP

Accesses the LDAP (*Lightweight Directory Access Protocol*) configuration environment.

Syntax:

```
Config>feature ldap
```

Example:

```
Config>feature ldap
-- LDAP User Configuration --
LDAP config>
```

2.4.14.25 FEATURE MAC-FILTERING

Accesses the MAC address packet *filtering* configuration environment.

Syntax:

```
Config>feature mac-filtering
```

Example:

```
Config>feature mac-filtering
-- MAC Filtering user configuration --
Filter config>
```

2.4.14.26 FEATURE MANAGEMENT

Accesses the *feature management* configuration menu. This feature allows you to schedule tasks to run upon receipt of an *NSLA advisor* (*Network Service Level Advisor*) or a system event notification. For more information about the NSLA feature, please see the following manual: *Teldat Dm754-I NSLA*.

Syntax:

```
Config>feature management
```

Example:

```
Config>feature management
-- Management user configuration --
MGT config>
```

The commands available at the feature's menu include:

```
MGT config>?
  no            Negate a command or set its defaults
  operation     Configure an operation to be executed
  script        Configure a script to be executed
  exit          Exit to parent menu
```

An operation is defined as a task that must be performed upon receipt of an advisor or system event notification. Configuring an operation is a two-phase process. In the first phase, you specify the task that you want to run and its possible parameters. In the second phase, you set up the advisor or the event whose notification will trigger the running operation.

A script is a list of commands that can be executed in an operation after an advisor notification is received, after an event occurs or once it is manually executed. The last option can be executed from the monitor menu under *feature management*. Multiple scripts execute sequentially in the order in which they were triggered. This means that two scripts cannot run at the same time.

Command history:

Release	Modification
11.00.05	The event option was introduced as of version 11.00.05.
11.01.01	The event option was introduced as of version 11.01.01.

2.4.14.26.1 operation <id> system reset

This command causes the device to *reset* when a notification is received from the advisor configured with the **operation <id> track nsla-advisor <advisor id>** command.

Example:

```
MGT config>operation 2 system reset
```

To delete an operation, use the **no operation <id>** command.

2.4.14.26.2 operation <id> system script <id_script>

Executes a list of commands configured on the system script (identified by <id_script>) and on an operation when a notification is received from the advisor or the event configured. This is possible with the **operation <id> track nsla-advisor <advisor id>** and **operation <id> track event <text>** command.

Example:

```
MGT config>operation 1 system script 3
```

To delete an operation, use the **no operation <id>** command.

Command history:

Release	Modification
11.00.05	Script execution and command script were introduced as of version 11.00.05.
11.01.00	Script execution and command script were introduced as of version 11.01.00.

2.4.14.26.3 operation <id> system event "event text"

Allows you to configure an event to be sent upon receipt of a notification from a configured advisor. If no advisor is configured, this command has no effect.

You can set custom text for the event.

Example:

```
MGT config$operation 1 system event "Text to show"
```

Thus, the event is sent when the advisor sends a notification to the operation:

```
06/02/16 22:34:16  SMGT.010 Text to show
```

Additional information:

To display the event, you need to enable SMGT subsystem events with INFO log level or above.

The event identifier is *SMGT.010* (as shown above).

Command history:

Release	Modification
11.00.05	The event option was introduced as of version 11.00.05.
11.01.01	The event option was introduced as of version 11.01.01.

2.4.14.26.4 operation <id> feature-power-switch reset <MTC+ id>

Causes the MTC+ to *reset* with the corresponding id when it receives a notification from the advisor configured with the **operation <id> track nsla-advisor <advisor id>** command.

Example:

```
MGT config>operation 1 feature-power-switch reset f
```

To delete an operation, use the **no operation <id> command**.

Command history:

Release	Modification
11.00.03	New command added.

2.4.14.26.5 operation <id> track nsla-advisor <advisor id>

This command is used when you want an operation to receive notifications from an *NSLA advisor.*

Example:

```
MGT config>operation 2 track nsla-advisor 1
```

To stop the operation receiving notifications from said advisor, use the **no operation <id> track** command.

2.4.14.26.6 operation <id> track event <text> {filter <id> text <text>}

You use this command when you want an operation to detect an *event* occurrence. You can either specify the name of the *event* to track (track all *events* with the specified name) or you can add filters to limit tracking to one or more text strings. To apply the desired filters, you need to use the *filter* option. This option allows you to configure one or more text filters. These text filters are applied as an OR operation.

Example:

```
MGT config>operation 1 track event IP.066 ?
  filter    Adds a filter
  <cr>
MGT config>operation 1 track event IP.066 filter 1 text "192.168.215.255"
```

To halt operation event tracking, use the **no operation <id> track** command .

Command history:

Release	Modification
11.01.03	The **track event** command was introduced as of version 11.01.03.

2.4.14.26.7 operation <id> name <text>

Adds a description to an operation for monitoring purposes.

Example:

```
MGT config>operation 1 name lte-up
```

To remove the description, use the **no operation <id> name** command.

Command history:

Release	Modification
11.00.05	The **name** command was introduced as of version 11.00.05.
11.01.00	The **name** command was introduced as of version 11.01.00.

2.4.14.26.8 operation <id> number-of-triggers <num>

Controls the number of actions that are performed when an Event or *NSLA advisor* notification is received.

Example:

```
MGT config>operation 1 number-of-triggers 1
```

To remove this command, use the **no operation <id> number-of-triggers** command.

Command history:

Release	Modification
11.01.03	The **number-of-triggers** command was introduced as of version 11.01.03.

2.4.14.26.9 script <id_script> commands <text>

Allows you to add commands to a script that will be run in an operation when a notification is sent from the advisor or when an event is registered in the system.

The first script command will execute from the GESTCON menu (i.e., the starting point that grants access to other processes).

Example:

```
MGT config>script 1 commands "p 4"
```

To remove the commands from the script, use the **no script <id_script>** command.

Note

Before designing your own script, please take the following considerations into account.

-The commands should be written as described in the manuals (i.e., literally). Using abbreviations for the commands entered is not recommended. For example: instead of writing "fea ntp" to access the NTP menu, enter "feature ntp". The use of abbreviations in this system can cause the script to behave in an unpredictable fashion.

-To navigate between different menus or access the console management process (**GESTCON**), you must use the "root" command. This action is equivalent to executing the CTRL+P shortcut in the console.

-The confirmation question asked by certain commands can be bypassed by entering the "yes" option at the end of the command (for instance, write 'script *n* commands "save yes"'). This speeds up script execution.

Command history:

Release	Modification
11.00.05	**script <id_script> commands** were introduced as of version 11.00.05.
11.01.00	**script <id_script> command**s were introduced as of version 11.01.00.

2.4.14.27 FEATURE NETFLOW

Accesses the *netflow* configuration environment.

Syntax:

```
Config>feature netflow
```

Example:

```
Config>feature netflow
NETFLOW config
```

For further information on how to configure *netflow*, please see the following manual: *Teldat Dm789-I NETFLOW* .

2.4.14.28 FEATURE NSLA

Accesses the *NSLA* (*Network Service Level Advisor*) configuration environment that provides functionalities for monitoring the service level offered by the network (Service Level) and for generating notifications related to SLAs *(Service Level Agreements)*.

Syntax:

```
Config>feature nsla
```

Example:

```
Config>feature nsla
-- Feature Network Service Level Advisor --
NSLA config>
```

For further information, please see the following manual: *Teldat Dm754-I NSLA (Network Service Level Advisor)* .

2.4.14.29 FEATURE NSM

Accesses the *NSM* (Network Service Monitor) system configuration environment that provides network service level information by using different probes built into the router that can measure performance.

Syntax:

```
Config>feature nsm
```

Example:

```
Config>feature nsm
-- Network Service Monitor configuration --
NSM config>
```

For further information, please see the following manual: *Teldat Dm749-I NSM* manual.

2.4.14.30 FEATURE NTP

Accesses the *NTP* (*Network Time Protocol*) configuration environment.

Syntax:

```
Config>feature ntp
```

Example:

```
Config>feature ntp
-- NTP Protocol user configuration --
NTP config>
```

For further information, please see the following manual: *Teldat Dm728-I NTP Protocol* .

2.4.14.31 FEATURE POLICY-MAP

Accesses the *policy-map* configuration environment.

Syntax:

```
Config>feature policy-map
```

Example:

```
Config>feature policy-map
-- Policy-Map Menu Configuration --
Policy-map Config>
```

For further information on how to configure *policy-map*, please see the following manual: *Teldat Dm795-I Policy Map Class Map*.

2.4.14.32 FEATURE POWER-SWITCH

Configures a test to determine whether an MTC+ is connected to a power supply.

Syntax:

```
Config$feature power-switch
```

Example:

```
Config$feature power-switch

-- POWER-SWITCH user configuration --

POWER-SWITCH$
```

For further information, please see the following manual: *Teldat Dm689-I MTC+*.

Command history:

Release	Modification
11.00.04	The "*power-switch*" feature was introduced as of version 11.00.04.

2.4.14.33 FEATURE PREFIX-LISTS

Accesses the *prefix lists* configuration environment.

Syntax:

```
Config>feature prefix-lists
```

Example:

```
Config>feature prefix-lists
-- Prefix Lists user configuration --
Prefix Lists config>
```

For further information on how to configure *prefix lists*, please see the following manual: *Teldat Dm780-I Prefix Lists* .

2.4.14.34 FEATURE RADIUS

Accesses the *RADIUS* protocol configuration environment.

Syntax:

```
Config>feature radius
```

Example:

```
Config>feature radius
-- RADIUS User Configuration --
RADIUS Config>
```

For further information on how to configure this protocol, please see the following manual: *Teldat Dm733-I RADIUS Protocol*.

2.4.14.35 FEATURE RMON

Accesses the *rmon* configuration environment.

Syntax:

```
Config>feature rmon
```

Example:

```
Config>feature rmon
-- Remote Network Monitoring configuration --
RMON config>
```

For further information on how to configure *rmon*, please see the following manual: *Teldat Dm796-I RMON Feature* .

2.4.14.36 FEATURE ROUTE-MAP

Accesses the *route map* configuration environment.

Syntax:

```
Config>feature route-map
```

Example:

```
Config>feature route-map
-- Route maps user configuration --
Route map config>
```

For further information on how to configure *route map*, please see the following manual: *Teldat Dm745-I Policy Routing*.

2.4.14.37 FEATURE SCADA-FORWARDER

Accesses the *SCADA forwarder* configuration environment.

Syntax:

```
Config>feature scada-forwarder
```

Example:

```
Config>feature scada-forwarder
-- SCADA Forwarder Configuration --
SCADA-FWD Cfg>
```

2.4.14.38 FEATURE SNIFFER

Accesses the built-in packet capture feature (*sniffer*) configuration environment.

Syntax:

```
Config>feature sniffer
```

Example:

```
Config>feature sniffer
-- SNIFFER configuration --
SNIFFER config>
```

2.4.14.39 FEATURE SPI

Accesses the *spi* configuration environment.

Syntax:

```
Config>feature spi
```

Example:

```
Config>feature spi
-- SPI global configuration --
SPI Config>
```

For further information on how to configure *spi*, please see the following manual: *Teldat Dm793-I SPI Agent* .

2.4.14.40 FEATURE SSH

Accesses the *ssh* configuration environment.

Syntax:

```
Config>feature ssh
```

Example:

```
Config>feature ssh
-- SSH protocol configuration --
SSH Config>
```

For further information on how to configure *ssh*, please see the following manual: *Teldat Dm787-I SSH Protocol* .

2.4.14.41 FEATURE STUN

Accesses the *STUN* client configuration environment.

Syntax:

```
Config>feature stun client
```

Example:

```
Config>feature stun client
STUN Client Config>
```

For further information, please see the following manual: *Teldat Dm769-I STUN Protocol.*

2.4.14.42 FEATURE SYSLOG

Accesses the *syslog* client configuration environment.

Syntax:

```
Config>feature syslog
```

Example:

```
Config>feature syslog
-- SYSLOG client configuration --
SYSLOG config>
```

For further information, please see the following manual: *Teldat Dm753-I Syslog Client* .

2.4.14.43 FEATURE TFTP

Accesses the *tftp* configuration environment.

Syntax:

```
Config>feature tftp
```

Example:

```
Config>feature tftp
-- TFTP user configuration --
TFTP config>
```

For further information on how to configure *tftp*, please see the following manual: *Teldat Dm765-I TFTP Protocol* .

2.4.14.44 FEATURE TMS

Accesses the *TMS* (Teldat Management System) configuration environment.

Syntax:

```
Config>feature tms
```

Example:

```
Config>feature tms
TMS config>
```

Command history:

Release	Modification
11.00.02	This command is obsolete as of version 11.00.02. The TMS feature is no longer supported.

2.4.14.45 FEATURE VLAN

Allows you to access the IEEE 802.1Q switch configuration environment to support the creation of virtual networks (*Virtual LAN*).

Syntax:

```
Config>feature vlan
```

Example:

```
Config>feature vlan
-- VLAN configuration --
VLAN config>
```

For further information, please see the following manual: *Teldat Dm751-I VLAN* .

2.4.14.46 FEATURE VLI

Accesses the *vli* configuration environment from devices that support it.

Syntax:

```
Config>feature vli
```

Example:

```
Config>feature vli
-- VLI configuration --
VLI config>
```

For further information on how to configure *vli*, please see the following manual: *Teldat Dm803-I Virtual Linux Interface (VLI)*.

2.4.14.47 FEATURE VRF

Accesses the *VRF* table (VPN Routing/Forwarding) configuration environment.

Syntax:

```
Config>feature vrf
```

Example:

```
Config>feature vrf
-- VRF user configuration --
VRF config>
```

For more information, please see the following manual: *Teldat Dm775-I VRF-Lite Facility.*

2.4.14.48 FEATURE WNMS

Accesses the *WNMS* (Wireless Network Management System) configuration environment.

Syntax:

```
Config>feature wnms
```

Example:

```
Config>feature wnms

-- Wireless Network Management System configuration --
WNMS config>
```

For further information on how to configure *WNMS*, please see the following manual: *Teldat Dm819-I Wireless Network Management System* .

Command history:

Release	Modification
11.00.03	The "*WNMS*" feature was introduced as of version 11.00.03.

2.4.14.49 FEATURE WRR-BACKUP-WAN

Accesses the *WRR* (WAN ReRoute) configuration environment.

Syntax:

```
Config>feature wrr-backup-wan
```

Example:

```
Config>feature wrr-backup-wan
-- WAN Reroute Backup user configuration --
Backup WRR>
```

For further information on this configuration environment, please see the following manual: *Teldat Dm727-I Backup WAN Reroute* .

2.4.14.50 FEATURE WRS-BACKUP-WAN

Accesses the *WRS* (WAN ReStoral) configuration environment.

Syntax:

```
Config>feature wrs-backup-wan
```

Example:

```
Config>feature wrs-backup-wan
-- WAN Back-up user configuration --
Back-up WAN>
```

For further information on how to configure *WRS backup*, please see the following manual: *Teldat Dm711-I Frame Relay ISDN Backup Configuration.* .

2.4.15 FILE

Accesses the files in the device's storage units.

The storage units are explicitly represented by a letter and the colon symbol (:). The flash memory unit is called **A:** and the smart card unit is called **S:**. Not all devices support both storage units. For more details, please see your device's installation manual.

One of the storage units is considered the active or default unit. To change the active storage unit, see the **config-media** command in this manual. If you want to refer to the active unit, do not include any unit names.

The **S:** unit is a compressed unit whereby the information that is stored is compressed with the gzip algorithm. To show this, the last letter is changed each time something is stored in the unit. For details on using this type of unit, please see your device's installation manual.

Syntax:

```
Config>file <operation> [parameters]
  copy      Copy files in the storage units
```

```
 create    Create a new file in the storage units
 delete    Delete files present in the device storage units
 format    Format a storage unit in the device
 list      Lists the files present in the storage units
 rename    Rename the files present in the device storage units
 type      Show files by console
```

- **< operation >** is the operation to be performed on the storage unit or file.
- **[parameters]** are the parameters required for the specified operation.

2.4.15.1 FILE COPY

Allows you to copy files to the storage units. The source and destination files can be on the same unit or on different ones. If they are from different units or they are not from the active unit, you have to specify the storage units. If both files belong to the active unit, you do not have to specify the storage unit.

Syntax:

```
Config>file copy <origin file> <destination file>
```

- **< origin file >** is the file you want to copy.
- **< destination file >** is the name of the destination file where you want to copy the specified source file.

Example 1:

```
Config>file copy xot1.cfg xot2.cfg
Config>
```

Example 2:

```
Config>file copy mike.cfg s:mike11.cfg
Config>
```

You can view the results of these two examples in the next section. Please note the MIKE11.CFG file appears as MIKE11.CFZ to indicate it is a compressed file.

2.4.15.2 FILE CREATE

Allows you to create files in the storage unit. You can create files with ASCII or HEX content. Press CTRL+P to end the file creation process.

Syntax:

```
Config>file create <destination file> <input mode> [yes]
```

- **< destination file >** is the name of the file you want to create.
- **< input mode >** is the type of input content used to create the specified file.
- the **yes** option can be used to automatically save the file in the system.

Example 1:

```
Config>file create example.txt ascii
Input File Content: (End CTRL+P)
Hi! This is a txt example file.
Save file(Yes/No)? Yes
File Successfully Saved

Config>
```

Example 2:

```
Config>file create examplehex.txt hex
Input File Content: (End CTRL+P)
0123456789abcdff0000000100000002000000000000000410000000000000000
0076e88f8b76cc1b62371bc1aa227130c63ca9614f7290bc56a52b8810eb4cc2
dcbc320e3f42b26f796ead2127e2d658495563afe35ac4aa149b0fef2207390d
467a95634d7592976b2f5bd92145cf1844912683bca7683fefaf6e0a8fa016dc
Save file(Yes/No)? Yes
File Successfully Saved

Config>
```

Use the **list** and **type** options to view the results of these examples and check whether the files have been correctly created.

Command history:

Release	Modification
11.00.04	This command option was introduced as of version 11.00.04.
11.01.06	The "[yes]" option was introduced as of version 11.01.06.

2.4.15.3 FILE DELETE

Allows you to delete files stored in the router's storage units. For security reasons, the file containing the device code cannot be deleted.

Syntax:

```
Config>file delete <filename>
```

- **< filename >** is the name of the file you want to delete.

Example 1:

```
Config>file list
Active Device: Flash
 A:   ROUTER.CFG         3510         12/09/02      12:45      Flash
 A:   TKR.CFG            1050         09/19/02      18:08      Flash
 A:   TEST.CFG           4708         04/26/02      15:33      Flash
 A:   SINTEST.CFG        4593         09/25/02      15:28      Flash
 A:   MIKE.CFG           1494         12/26/02      16:47      Flash
 A:   MIKE2.CFG          6302         12/13/02      10:09      Flash
 A:   XOT1.CFG           1494         12/26/02      14:33      Flash
 A:   XOT2.CFG           1494         12/27/02      12:27      Flash
 A:   XOT3.CFG           1554         12/26/02      13:18      Flash
 A:   APPCODE1.BIN       2760544         01/03/03      10:39      Flash

Flash Available Space : 2496 Kbytes

 S:   ROUTER.CFZ          802                                  SmartCard

SmartCard Available Space : 14400 bytes
Config>file delete s:router.cfz
Config>file list
Active Device: Flash
 A:   ROUTER.CFG         3510         12/09/02      12:45      Flash
 A:   TKR.CFG            1050         09/19/02      18:08      Flash
 A:   TEST.CFG           4708         04/26/02      15:33      Flash
 A:   SINTEST.CFG        4593         09/25/02      15:28      Flash
 A:   MIKE.CFG           1494         12/26/02      16:47      Flash
 A:   MIKE2.CFG          6302         12/13/02      10:09      Flash
 A:   XOT1.CFG           1494         12/26/02      14:33      Flash
 A:   XOT2.CFG           1494         12/27/02      12:27      Flash
 A:   XOT3.CFG           1554         12/26/02      13:18      Flash
 A:   APPCODE1.BIN    2760544         01/03/03      10:39      Flash

Flash Available Space : 2496 Kbytes
SmartCard Available Space : 15300 bytes
Config>
```

Example 2:

```
Config>file delete appcode1.bin
CLI Error: Application code files can not be deleted
CLI Error: Command error
Config>
```

2.4.15.4 FILE FORMAT

Allows you to format a storage unit on the device. At present, only the smart card can be formatted. Please note that formatting a unit will erase all the files on the unit.

Syntax:

```
Config>file format <store unit>
```

- **< store unit >** is the name of the storage unit you want to format. At present, only the smart card can be formatted.

Example:

```
Config>file format smartcard
Formatting, please wait ... OK
Config>
```

2.4.15.5 FILE LIST

Lists the files on the router's storage units. It also shows which unit is active. To change the active unit, see the **config-media** command in this manual.

Each line shows the following information: the unit identifier, filename, extension, byte size, date and time the unit was created, and the name of the storage unit as a text. Finally, the amount of free space available on each unit is shown.

Syntax:

```
Config>file list
```

Example:

```
Config>file list
Active Device: Flash
 A:   ROUTER.CFG              3510      12/09/02      12:45      Flash
 A:   TKR.CFG                 1050      09/19/02      18:08      Flash
 A:   TEST.CFG                4708      04/26/02      15:33      Flash
 A:   SINTEST.CFG             4593      09/25/02      15:28      Flash
 A:   MIKE.CFG                1494      12/26/02      16:47      Flash
 A:   MIKE2.CFG               6302      12/13/02      10:09      Flash
 A:   XOT1.CFG                1494      12/26/02      14:33      Flash
 A:   XOT2.CFG                1494      12/27/02      12:27      Flash
 A:   XOT3.CFG                1554      12/26/02      13:18      Flash
 A:   APPCODE1.BIN         2760544      01/03/03      10:39      Flash

Flash Available Space : 2496 Kbytes


 S:   MIKE11.CFZ               802                               SmartCard

SmartCard Available Space : 14400 bytes
Config>
```

Keep in mind that the smart card is a slow storage unit and may take several seconds to respond.

2.4.15.6 FILE RENAME

Allows you to rename files in the device's storage units. If you want to rename a file, you need to specify the original name first, followed by the new name. The unit indicated in the original name and the new name must match. For security reasons, the file containing the device code cannot be renamed.

Syntax:

```
Config>file rename <filename> <new name>
```

- **< filename >** is the name of the file you want to rename.
- **< new name >** is the new name you want to give to the specified file.

Example 1:

```
Config>file list
Active Device: Flash
 A:   ROUTER.CFG         3510         12/09/02      12:45      Flash
 A:   TKR.CFG            1050         09/19/02      18:08      Flash
 A:   TEST.CFG           4708         04/26/02      15:33      Flash
 A:   SINTEST.CFG        4593         09/25/02      15:28      Flash
 A:   MIKE.CFG           1494         12/26/02      16:47      Flash
 A:   MIKE2.CFG          6302         12/13/02      10:09      Flash
 A:   XOT1.CFG           1494         12/26/02      14:33      Flash
 A:   XOT2.CFG           1494         12/27/02      12:27      Flash
 A:   XOT3.CFG           1554         12/26/02      13:18      Flash
 A:   APPCODE1.BIN    2760544         01/03/03      10:39      Flash

Flash Available Space : 2496 Kbytes

 S:   ROUTER.CFZ          802                                  SmartCard

SmartCard Available Space : 14400 bytes
Config>file rename s:router.cfz s:backup.cfz
Config>file list
Active Device: Flash
 A:   ROUTER.CFG         3510         12/09/02      12:45      Flash
 A:   TKR.CFG            1050         09/19/02      18:08      Flash
 A:   TEST.CFG           4708         04/26/02      15:33      Flash
 A:   SINTEST.CFG        4593         09/25/02      15:28      Flash
 A:   MIKE.CFG           1494         12/26/02      16:47      Flash
 A:   MIKE2.CFG          6302         12/13/02      10:09      Flash
 A:   XOT1.CFG           1494         12/26/02      14:33      Flash
 A:   XOT2.CFG           1494         12/27/02      12:27      Flash
 A:   XOT3.CFG           1554         12/26/02      13:18      Flash
 A:   APPCODE1.BIN       2760544         01/03/03      10:39      Flash

Flash Available Space : 2496 Kbytes

 S:   BACKUP.CFZ          802                                  SmartCard

SmartCard Available Space : 14400 bytes
Config>
```

Example 2:

```
Config>file rename appcode1.bin appcode1_bak.bin
CLI Error: Application code files can not be renamed
CLI Error: Command error
Config>
```

Example 3:

```
Config>file rename s:router.cfz a:router.cfg
Disk Units do not match
Config>
```

2.4.15.7 FILE TYPE

Displays the files stored in the device's storage units. Each non-printable character is replaced by a dot (.) in the display.

Syntax:

```
Config>file type [header] <filename> [hex | text]
```

- **header** this option only shows the first few lines of the file.
- **< filename >** is the name of the file you want to view.
- **hex** dumps the file content byte by byte in hexadecimal mode, with its correspondence in text characters. Codes that do not correspond to standard text characters appear as a dot (.) in the text correspondence.
- **text** displays the contents of the file in text mode. Non-standard characters are replaced by a dot (.) in the display.

If no option is specified, the entire file is displayed in text mode.

Example 1:

```
Config>file type header temp.cfg
; Showing System Configuration for access-level 0 ...
; C4i IPSec Router 1 16 Version 10.6.27TM
log-command-errors
no configuration
set data-link astm serial0/0
```

```
;
protocol ip
; -- Internet protocol user configuration --
   internal-ip-address 172.24.78.116
;
   address ethernet0/0 172.24.78.116 255.255.0.0
;
;
;
exit
;
;
protocol bgp
; -- Border Gateway Protocol user configur
Config>
```

Example 2:

```
Config>file type temp.cfg
; Showing System Configuration for access-level 0 ...
; C4i IPSec Router 1 16 Version 10.6.27TM
log-command-errors
no configuration
set data-link astm serial0/0
;
protocol ip
; -- Internet protocol user configuration --
   internal-ip-address 172.24.78.116
;
   address ethernet0/0 172.24.78.116 255.255.0.0
;
;
;
exit
;
;
protocol bgp
; -- Border Gateway Protocol user configuration --
   enable
;
   aggregate default 192.168.0.0 mask 255.255.0.0
   aggregate default 10.0.0.0 mask 255.0.0.0
;
   aggregate 10.0.0.0 mask 255.0.0.0 10.0.0.0 mask 255.0.0.0 refines
;
   as 100
exit
;
dump-command-errors
end
; --- end ---

Config>
```

Example 3:

```
Config>file type header temp.cfg hex
3b 20 53 68 6f 77 69 6e 67 20 53 79 73 74 65 6d ; ; Showing System
20 43 6f 6e 66 69 67 75 72 61 74 69 6f 6e 20 66 ;  Configuration f
6f 72 20 61 63 63 65 73 73 2d 6c 65 76 65 6c 20 ; or access.level
30 20 2e 2e 2e 0d 0a 3b 20 43 34 69 20 49 50 53 ; 0 .....; C4i IPS
65 63 20 52 6f 75 74 65 72 20 31 20 31 36 20 56 ; ec Router 1 16 V
65 72 73 69 6f 6e 20 31 30 2e 36 2e 32 37 54 4d ; ersion 10.6.27TM
0d 0a 0d 0a 6c 6f 67 2d 63 6f 6d 6d 61 6e 64 2d ; ....log.command.
65 72 72 6f 72 73 20 0d 0a 6e 6f 20 63 6f 6e 66 ; errors ..no conf
69 67 75 72 61 74 69 6f 6e 20 0d 0a 73 65 74 20 ; iguration ..set
64 61 74 61 2d 6c 69 6e 6b 20 61 73 74 6d 20 73 ; data.link astm s
65 72 69 61 6c 30 2f 30 0d 0a 3b 0d 0a 70 72 6f ; erial0.0..;..pro
```

```
74 6f 63 6f 6c 20 69 70 0d 0a 3b 20 2d 2d 20 49 ; tocol ip..; .. I
6e 74 65 72 6e 65 74 20 70 72 6f 74 6f 63 6f 6c ; nternet protocol
20 75 73 65 72 20 63 6f 6e 66 69 67 75 72 61 74 ;  user configurat
69 6f 6e 20 2d 2d 0d 0a 20 20 20 69 6e 74 65 72 ; ion ....   inter
6e 61 6c 2d 69 70 2d 61 64 64 72 65 73 73 20 31 ; nal.ip.address 1
37 32 2e 32 34 2e 37 38 2e 31 31 36 0d 0a 3b 0d ; 72.24.78.116..;.
0a 20 20 20 61 64 64 72 65 73 73 20 65 74 68 65 ; .   address ethe
72 6e 65 74 30 2f 30 20 31 37 32 2e 32 34 2e 37 ; rnet0.0 172.24.7
38 2e 31 31 36 20 32 35 35 2e 32 35 35 2e 30 2e ; 8.116 255.255.0.
30 0d 0a 3b 0d 0a 3b 0d 0a 3b 0d 0a 65 78 69 74 ; 0..;..;..;..exit
0d 0a 3b 0d 0a 3b 0d 0a 70 72 6f 74 6f 63 6f 6c ; ..;..;..protocol
20 62 67 70 0d 0a 3b 20 2d 2d 20 42 6f 72 64 65 ;  bgp..; .. Borde
72 20 47 61 74 65 77 61 79 20 50 72 6f 74 6f 63 ; r Gateway Protoc
6f 6c 20 75 73 65 72 20 63 6f 6e 66 69 67 75 72 ; ol user configur

Config>
```

2.4.16 FIRMWARE-CHECKING

Enables the firmware integrity check. This command is used in conjunction with the **backup-files** command to ensure device availability in serious cases of file system corruption.

Note

This command is only effective on devices with partitioned flash memory.

Syntax:

```
Config> firmware-checking
```

Example:

```
Config>firmware-checking
Config>
```

2.4.17 FIXED-NUMBER-SNMP

This feature sets the persistence of bandwidth-reservation (BRS) interface and class indexes over time, regardless of whether new interfaces/classes are added or existing ones deleted. With this new functionality, the order of interfaces and classes will match the order of creation (i.e., new interfaces and classes will be added at the end of their corresponding lists, instead of being grouped according to their priority). The order of interfaces and classes will remain the same when saved, even after the device is restarted.

The physical interfaces activated by the router's license always appear first and in a fixed position.

This functionality is enabled at the fixed-number-snmp menu

Syntax:

```
Config>fixed-number-snmp
```

Example:

```
Config>fixed-number-snmp
-- Fixed configuration config>
```

For further information on how the *fixed-number-snmp* feature works, please see the following manuals: *Teldat Dm715-I Bandwidth Reservation System* and *Teldat Dm772-I Configuration Interface* .

Command history:

Release	Modification
11.01.08	The "*fixed-number-snmp*" command was introduced as of version 11.01.08.
11.01.09	The "*fixed-number-snmp*" command was modified as of version 11.01.09.

2.4.18 FORMAT

Formats the specified storage device.

Syntax:

```
Config>format <device>
```

- **< device >** is the name of the storage device to be formatted. At present, you can only format the smart card system.

Example:

```
Config>format smartcard

Formatting, please wait ... OK
Config>
```

2.4.19 GLOBAL-PROFILES

Defines router profiles (PPP, ATM, etc.).

Syntax:

```
Config>global-profiles <profile name>

  dial          Access the DIAL profiles configuration environment
  ppp           Access the PPP profiles configuration environment
  tcp-menu      Access the TCP profiles configuration environment
  trmtp-menu    Access the TRMTP profiles configuration environment
```

- **< profile name >** is the ID of the profile type whose configuration menu you want to access.

Command history:

Release	Modification
11.00.02	Options **tcp-menu** and **trmtp-menu** are obsolete as of version 11.00.02. The DEP protocol is no longer supported.

2.4.19.1 GLOBAL-PROFILES DIAL

Accesses the dial profile configuration environment.

Syntax:

```
Config>global-profiles dial
```

Example:

```
Config>global-profiles dial
-- Dial Profiles Configuration --
Dial Profiles config>
```

This configuration environment is described in greater detail in the following manual: *Teldat Dm732-I Dial Profile* .

2.4.19.2 GLOBAL-PROFILES PPP

Accesses the *PPP* (Point-to-Point Protocol) profile configuration environment.

Syntax:

```
Config>global-profiles ppp
```

Example:

```
Config>global-profiles ppp
-- PPP Profiles Configuration --
PPP Profiles config>
```

This configuration environment is described in greater detail in the following manual: *Teldat Dm710-I PPP Interface* .

2.4.19.3 GLOBAL-PROFILES TCP-MENU

Accesses the DEP over TCP (*Transport Control Protocol*) profile configuration environment for the encapsulation of POS or Dataphone traffic in IP networks.

For further information on this configuration environment, please see the following manual: *Teldat Dm263-I Forwarder DEP*.

Syntax:

```
Config>global-profiles tcp-menu
```

Example:

```
Config>global-profiles tcp-menu
-- UDAFO TCP Configuration Menu --
UDAFO TCP Cfg>
```

Command history:

Release	Modification
11.00.02	This command is obsolete as of version 11.00.02. The DEP protocol is no longer supported.

2.4.19.4 GLOBAL-PROFILES TRMTP-MENU

Accesses the DEP over the *TRMTP* (Trivial Message Transfer Protocol) profile configuration environment for the encapsulation of POS or Dataphone traffic in IP networks.

For further information on this configuration environment, please see the following manual: *Teldat Dm263-I Forwarder DEP*.

Syntax:

```
Config>global-profiles trmtp-menu
```

Example:

```
Config>global-profiles trmtp-menu
-- UDAFO TRMTP Configuration Menu --
UDAFO TRMTP Cfg>
```

Command history:

Release	Modification
11.00.02	This command is obsolete as of version 11.00.02. The DEP protocol is no longer supported.

2.4.20 LICENCE-CHANGE

Allows you to change the device license in order to enable/disable certain features. You need a special license to be able to use this command. After you have selected the language that will be used when interacting with the device, the current license appears. Once the type of installation performed has been specified, you must decide whether to enable each of the available functionalities in this new license. Once you have finished doing this, you are asked whether you want to save the changes. If you do, you are asked whether you want to restart the device immediately. If you prefer to perform this operation later, the changes will not be effective until the reboot takes place.

Syntax:

```
Config>licence-change
```

Example:

```
Config>licence-change

1. English
2. Español
Language/Idioma[1]? 1

Current licence: 1 249 C4i_AdHoc IPSec SNA
```

```
Last executed task was Register

FUNCTIONALITY   ENABLED
--------------  ----------
ISDN            NO
IPSEC           YES
SNA             YES
VOIP            NO

Please indicate here if you are installing the router to carry out one of the fol
lowing tasks:
   R. Register for the first time
   M. Maintenance to resolve an event
Please select an option > m
Enable ISDN (Yes/No)[Y]?y
Enable IPSEC (Yes/No)[Y]?y
Enable SNA (Yes/No)[Y]?y
Enable VOIP (Yes/No)[Y]?y
Enable NOE (Yes/No)[Y]?n

Executed task was Maintenance
FUNCTIONALITY   ENABLED
--------------  ----------
ISDN            YES
IPSEC           YES
SNA             YES
VOIP            YES
NOE             NO
Do you want to save changes (Yes/No)[N]?y
Searching licence code for new configuration...
New licence established: 1 261 C4i_AdHoc ISDN IPSec SNA VoIP
You must restart/reload for the changes to take effect
Are you sure you want to reload the device (Yes/No)[N]?n
Config>
```

2.4.21 LIST

Displays configuration information on the active storage unit (flash or smart card), protocols, interfaces, users, pools and enabled patches.

Syntax:

```
Config>list <info>
  configuration    List generic configuration information
  devices          List router devices
  patch            Check the personalized parameters that are active
  pool             Number of bytes assigned to each memory pool
  user             Displays the list of registered users
```

- **< info >** is the identifier of the information you want listed.

Command history:

Release	Modification
11.00.06	The *pool* option is obsolete as of version 11.00.06.
11.01.01	The *pool* option is obsolete as of version 11.01.01.

2.4.21.1 LIST CONFIGURATION

Lists configuration information about the active device.

Syntax:

```
Config>list configuration
```

Example:

```
Router Config>list configuration
```

```
Hostname: Router
Contact person: ......
Host Location: .......
No console authentication
No Telnet authentication
No FTP access authentication
Configurable protocols:
Num    Name         Protocol
 0     IP           DOD-IP
 3     ARP          Address Resolution Protocol
 4     H323         H323
 6     DHCP         Dynamic Host Configuration Protocol
10     BGP          BGP
11     SNMP         SNMP
12     OSPF         Open SPF-Based Routing Protocol
13     RIP          Route Information Protocol
17     SIP          SIP
23     ASRT         Adaptive Source Routing Transparent Enhanced Bridge
25     NHRP         Next Hop Resolution Protocol
26     DLS          Data Link Switching
29     L2TP         L2TP
30     EAPOL        Extensible Authentication Protocol Over LAN
31     Preauth      WLAN Preauthentication

713 bytes of memory used for configuration
Router Config>
```

2.4.21.2 LIST DEVICES

Lists information about the device's available/configured interfaces.

Syntax:

```
Config>list devices
```

Example:

```
Config>list devices
Interface            Connector     Type of interface
ethernet0/0          GE0/FE0/LAN1  Fast Ethernet interface
ethernet0/1          GE1/FE1/LAN2  Fast Ethernet interface
bri0/0               BRI/ISDN1     ISDN Basic Rate Int
x25-node             ---           Router->Node
Config>
```

2.4.21.3 LIST PATCH

Checks the customizable parameters that are active.

Syntax:

```
Config>list patch
```

Example:

```
Config>list patch
Patch Name                    value
-----------------------------------------
ARPI_SND_LCL                      1 (0x1)
Config>
```

2.4.21.4 LIST POOL

Shows the number of bytes assigned to each memory pool as well as the number of free bytes.

Syntax:

```
Config>list pool
```

Example:

```
Config>list pool
3 Iorbs pool: 4194304
4 MSGs pool: 1204000
Total memory pools: 5398304 Total free memory: 0
Config>
```

Command history:

Release	Modification
11.00.06	The "*list pool*" command is obsolete as of version 11.00.06.
11.01.01	The "*list pool*" command is obsolete as of version 11.01.01.

2.4.21.5 LIST USER

Displays the list of registered users, their password, access level, mode of access and whether or not they are enabled.

Syntax:

```
Config>list user
```

Example:

```
Config>list user
Name                 Password              Access Level   Strict  Enabled
config               ****************      [10]Config       N        N
monitor              ****************      [ 5]Monitor      N        Y
root                 ****************      [15]Root         N        Y
mabm                 ****************      [15]Root         N        Y
guest                ****************      [ 2]             Y        Y
viewer               ****************      [ 1]Events       N        Y
Config>
```

2.4.22 LOG-COMMAND-ERROR

Initializes (clears) the error log holding the errors that have occurred when running commands from the configuration console.

Syntax:

```
Config>log-command-errors
```

Example:

```
Config>log-command-errors
Config>
```

This command is usually used before loading a new configuration file because you can follow it with the the **dump-command-errors** command to view any possible errors.

2.4.23 MANAGEMENT

Enters the master router configuration environment.

Syntax:

```
Config>management
```

Example:

```
Config>management
-- Routers management user configuration --
Management config>
```

2.4.24 NETWORK

Allows you to access the command menu to configure a specific interface. To exit this menu, type **exit**.

Syntax:

```
Config>network <name>
```

- Where **< name >** is the interface name.

You can find out which interfaces are available on the device by typing **list devices**.

Example 1:

```
Config>network ethernet0/0
-- Ethernet Interface User Configuration --
ethernet0/0 config>
```

Example 2:

```
Config>network serial0/4
CLI Error: Unrecognized command or invalid value
Config>
```

For more information on interfaces, please see the *Teldat Dm722-I Telephony Over IP* manual and the manual associated with the network interface whose configuration environment you want to access.

2.4.25 NO

Sets parameters back to their default values, disables options or deletes previously added configuration elements.

Syntax:

```
Config>no <command> [parameters]
```

- **< command >** is the name of the command you want undone.
- **[parameters]** are the parameters required for the specified command.

2.4.25.1 NO ADD DEVICE

Deletes the specified virtual interface. (For an alternative means of deleting an interface, please see **no device**).

Syntax:

```
Config>no add device <virtual interface type> [options]
```

- **< virtual interface type>** is the virtual interface you want deleted.
- **[options]** The options depend on the type of virtual interface you are going to delete. They are the same as those used to create the interface with the **add device** command.

Example:

```
Config>no add device ppp 1
Config>
```

Release	Modification
11.01.00	The new command was introduced as of version 11.01.00.

2.4.25.2 NO AUTOINSTALL

Deletes the configuration of the autoinstall parameters.

Syntax:

```
Config>no autoinstall
```

Example:

```
Config>no autoinstall
Config>
```

Release	Modification
11.00.03	This command is obsolete as of version 11.00.03.

2.4.25.3 NO BANNER

Deletes the specified type of banner.

Syntax:

```
Config>no banner <type>
```

< type > specifies the type of banner to be deleted. Currently, the only type available is the access (login) banner.

Example:

```
Config>no banner login
Config>
```

2.4.25.4 NO CONFIGURATION

Deletes all existing configurations.

Syntax:

```
Config>no configuration
```

Example:

```
Config>no configuration
Config>
```

2.4.25.5 NO CONFIRM-CFG

Rejects the current test configuration, thus causing the device to restart with the previous configuration following a warning message. For more information, see the **confirm-cfg-needed** and **confirm-cfg** commands.

Syntax:

```
Config>no confirm-cfg
```

Example:

```
Config>no confirm-cfg
!!!!LOOK OUT!!!!
This command reboots the system
If you go on, previous configuration will be restored
You can test this configuration again with: set file-cfg TEMP
To go on write RECOVER: RECOVER
```

You can also perform this action via SNMP. See *CONFIRM-CFG-NEEDED* on page 36.

2.4.25.6 NO CONFIRM-CFG-NEEDED

Disables the requirement that new configurations be confirmed. See **confirm-cfg-needed**.

Syntax:

```
Config>no confirm-cfg-needed
```

Example:

```
Config>no confirm-cfg-needed
Config>
```

You can also perform this action via SNMP. See *CONFIRM-CFG-NEEDED* on page 36.

2.4.25.7 NO CONTACT-PERSON

Deletes the contact name/ID. For more information, see the **set-contact-person** command.

Syntax:

```
Config>no contact-person
```

Example:

```
Config>no contact-person
Config>
```

2.4.25.8 NO DESCRIPTION

Clears the device configuration description(s). For more information, see the **description** command.

Syntax:

```
Config>no description
```

Example:

```
Config>no description
Config>
```

2.4.25.9 NO DEVICE

Deletes the specified virtual interface. See the **no add device** command for an alternative way to delete an interface.

Syntax:

```
Config>no device <name>
```

- **< name>** is the name of the interface to be deleted.

Example:

```
Config>no device fr1
Config>
```

2.4.25.10 NO FIRMWARE-CHECKING

Disables checks when working with firmware files.

Syntax:

```
Config>no firmware-checking
```

Example:

```
Config>no firmware-checking
Config>
```

2.4.25.11 NO HOST-LOCATION

Deletes the text indicating the device's location. For more information, see the **set-host-location** command.

Syntax:

```
Config>no host-location
```

Example:

```
Config>no host-location
Config>
```

2.4.25.12 NO HOSTNAME

Deletes the assigned device name. For more information, see the **set hostname** command.

Syntax:

```
Config>no hostname
```

Example:

```
Config>no hostname
Config>
```

2.4.25.13 NO LOGIN

Disables the **login** options configured using the **set login** command.

Syntax:

```
Config>no login <option>
```

- **<option>** is the login option to disable. Currently, only the case-sensitive option (which is used to disable case checking) is supported.

Example:

```
Config> no login case-sensitive
Config>
```

2.4.25.14 NO PASSWORD

Deletes the device's password settings. For more information, see the **setpassword** command.

Syntax:

```
Config>no password
```

Example:

```
Config>no password
Config>
```

2.4.25.15 NO PRIVILEGE

Deletes the settings entered with the **privilege** command.

Syntax:

```
Config>no privilege <access-level> [<command-path>]
```

- **<access-level>** is the access level to which the command applies.
- **<command-path>** this is an optional parameter. If specified, only the command corresponding to this command path is deleted. If it is not specified, all **privilege** commands relating to the specified access level are deleted.

Example:

```
Config>no privilege 3
Config>
```

2.4.25.16 NO RUSH-ENGINE

Disables **rush engine** or restores the default idle timeout configuration, which is **5** seconds.

Syntax:

```
Config>no rush-engine [timeout]
```

- **timeout** is an optional parameter. If specified, the idle timeout is set to default. If you do not specify this parameter, **rush engine** is disabled.

Example:

```
Config>no rush-engine timeout
Config>
```

Command history:

Release	Modification
11.01.00	This command was introduced as of version 11.01.00.

2.4.25.17 NO USER

Deletes a user from the user list. You can delete as many users as you want, but not the last root user if there are still some users registered in the system. In this case, you can only delete the other registered users (were you to actually delete the last root user without removing the other registered users, then you wouldn't be able to manage those users). You can delete the last root user once you have removed all the registered users. Then the system would no longer request a username and password to access the device because there would be no users left in the system.

Syntax:

```
Config>no user <user-name>
```

- **< user-name >** is the registered user's name.

Example:

```
Config>no user mabm
Config>
```

2.4.26 NODE

Allows you to access node configuration (X.25, XOT and 270). You can also gain access by typing the network command followed by the interface on which the node is configured.

Syntax:

```
Config>node <name>
  270    Access the 270 configuration
  x25    Access the X25 node configuration
  xot    Access the XOT configuration
```

- **< name >** is the name of the node whose configuration menu you want to access.

There are three types of configurable nodes:

2.4.26.1 270 NODE

Accesses the 270 node configuration environment.

Syntax:

```
Config>node 270
```

Example:

```
Config>node 270
270 Config>
```

2.4.26.2 X25 NODE

Accesses the X.25 node configuration environment.

Syntax:

```
Config>node x25
```

Example:

```
Config>node x25

X25 Config>
```

For more information on the X.25 node configuration environment, please see the following manual: *Teldat Dm707-I X.25*.

2.4.26.3 XOT NODE

Accesses the XOT (X.25 over TCP/IP) node configuration environment.

Syntax:

```
Config>node xot
```

Example:

```
Config>node xot
XOT config>
```

For more information on the XOT node configuration environment, please see the following manual: *Teldat Dm713-I XOT Protocol*.

2.4.27 PRIVILEGE

Allows you to define custom execute permissions for the device's process commands.

Syntax:

```
Config>privilege <level> <command path> [all]
```

- **< level >** is the access level to assign to the command specified in <*command path*>.
- **< command path>** is the command path.
- **< all >** is an option that lets you apply the specified access level to all subcommands (paths match) of the command specified in <*command path*>.

The <**level**> parameter, which specifies the level of access to be assigned to the command, allows values in the range of 0-15.

The **<command path>** parameter specifies the specific path of the command whose access level we want to customize. It is built by typing the commands necessary to execute the command consecutively one after another and separated by a **>**.

It must be enclosed in double quotation marks (“) and its syntax must conform to the following standards:

1) As a general rule, to define an access level for a specific command, the path identifying it must start with a **>**.

Example:

```
Config>privilege 12 “>monitor>network ppp1”
Config>
```

To assign an access level to a command regardless of the current menu, we will have to omit character **>** and not use it within the path.

Example:

```
Config$privilege 7 "list"
Config$privilege 7 "ppp>ipcp"
CLI Error: Wrong specification of command path (misuse of >)
CLI Error: Command error
Config$
```

The **>config** path refers to both static and dynamic configuration. Adding different permissions to the same command will depend on whether the user is configuring the device statically or dynamically.

2) When specifying the command path, the greater-than character (**>**) is used as a separator between the various device process levels and menus.

Example:

```
Config$privilege 12 ">config>network ppp1>ppp>ipcp"
Config$
```

3) While defining the command path, you can use an asterisk (*****) as a wildcard parameter.

Example:

```
Config$privilege 5 ">config>network *>list"
Config$
```

This would assign level 5 access to the **list** command on any net configuration menu (*net ppp1, net ppp2, net fr1, etc.).*

You can run the **show** configuration command to help you find out the full path of a device parameter configuration command.

Example:

```
Config>show config
; Showing System Configuration for access-level 15 ...

log-command-errors
no configuration
set inactivity-timer disabled
add device ppp 1
set data-link sync serial0/0
set data-link x25 serial0/1
;
network ethernet0/0
; -- Ethernet Interface User Configuration --
   ip address 172.24.73.23 255.255.0.0
```

```
;
;
;
;
exit
;
;
;
network ethernet0/1
; -- Ethernet Interface User Configuration --
   no ip address
;
exit
;
;
;
;
network x25-node
; -- X25-node interface configuration --
   no ip address
;
exit
;
;
network ppp1
; -- Generic PPP User Configuration --
   ip address 1.1.1.1 255.255.255.0
;
;
;
;
   ppp
; -- PPP Configuration --
      ipcp remote address fixed 1.1.1.2
   exit
;
   base-interface
; -- Base Interface Configuration --
      base-interface serial0/0 link
;
   exit
;
exit
;
;
;
dump-command-errors
end
; --- end ---
Config>
```

If, for example, you want to grant level 12 access to the assignment of a base interface to the ppp1 net (**base-interface serial0/0 link command**), you need to use the following path: "**>config>network ppp1>base-interface>base-interface serial0/0 link**“

The **<all>** option allows you to apply the specified level to all subcommands specified in **<command path>**. That is, to all those commands whose path matches from the start (allowing for wildcards).

Example:

```
Config>privilege 12 “>config>network” all
Config>
```

This configuration applies the access level to all commands whose path begins with "**>network**", such as ">**config>network ppp1**" or "**>network ppp2>ppp**".

If you do not specify this option, the access level *only* applies to the command defined by the path, that is, to the one whose execution ends with a carriage return <CR>. So, for example, if we have the following configuration (without the **all** option):

```
Config>privilege 12 “>config>confirm-cfg”
Config>privilege 12 “>config>protocol *>no *"
Config>
```

And we access as a level 10 user and try to run the following commands:

```
Config>confirm-cfg
CLI Error: Incomplete command
Config>protocol ip
-- Internet protocol user configuration --
IP config>no description
CLI Error: Command error
IP config>
```

We get an execution error because these commands require level 12 access. If, however, we run:

```
Config>confirm-cfg-needed default
Config>protocol ip
-- Internet protocol user configuration --
IP config>no aggregation-route 192.168.0.0 255.255.0.0
IP config>
```

We don't get an error this time because the **all** option is not configured. The change of access level does not affect these commands, which have level 10 access by default.

The systems lets you define multiple access levels for the same command, and you have the option of creating two user types: *default* and *strict* (see the **user** command). If this situation occurs and you access the system using a default user (not *strict*), **the command acquires the highest access level**.

2.4.27.1 Example: user to configure IP telephony

Let's take a look at how to use command execute permissions to define a user who is only allowed to configure parameters related to the device's IP telephony functionality. Two users are defined: one with root privileges and the other with level 7 access, monitoring privileges and other privileges configured with the **privilege** command.

The VoIP user is allowed to access all the VoIP voice interfaces, the **telephony** menu, the **sip** and **h323** protocols; configure **access list 50** to use it in telephony; configure **two NSM operations**, **two filters**, **alarms** and **NSLA advisors**, and one **global-profiles dial** profile called VoIP to apply on an ISDN voice interface; exit any menu using the **exit** command; and **save the configuration**. In this way, the user has complete control over the IP telephony configuration of the device, but he cannot alter any routing parameter values nor restart the device.

```
; -- Privilege Configuration –
privilege 7 ">config>feature access-list>access-list 50" all
privilege 7 ">config>feature access-list>access-list 51" all
privilege 7 ">config>feature access-list>no access-list 50" all
privilege 7 ">config>feature access-list>no access-list 51" all
privilege 7 ">config>feature nsla>advisor 10" all
privilege 7 ">config>feature nsla>advisor 11" all
privilege 7 ">config>feature nsla>alarm 10" all
privilege 7 ">config>feature nsla>alarm 11" all
privilege 7 ">config>feature nsla>filter 10" all
privilege 7 ">config>feature nsla>filter 11" all
privilege 7 ">config>feature nsla>no advisor 10" all
privilege 7 ">config>feature nsla>no advisor 11" all
privilege 7 ">config>feature nsla>no alarm 10" all
privilege 7 ">config>feature nsla>no alarm 11" all
privilege 7 ">config>feature nsla>no filter 10" all
privilege 7 ">config>feature nsla>no filter 11" all
privilege 7 ">config>feature nsm>no operation 10" all
privilege 7 ">config>feature nsm>no operation 11" all
privilege 7 ">config>feature nsm>operation 10" all
privilege 7 ">config>feature nsm>operation 11" all
privilege 7 ">config>global-profiles dial>no profile voip" all
privilege 7 ">config>global-profiles dial>profile voip" all
privilege 7 ">config>network voip" all
privilege 7 ">config>protocol h323" all
privilege 7 ">config>protocol sip" all
privilege 7 ">config>save" all
privilege 7 ">config>telephony" all
privilege 7 "exit"
```

```
;
;
user root hash-password A44AD55CE197114B241EE3DDEBB04660
;
user voip hash-password 7A325D20A3B026A12D094C61DB21D880
user voip access-level 7
;
event
; -- ELS Config --
   enable syslog subsystem CNSL ALL
   console
; --  Console Events Configuration  --
      log source-ip
      log prompt
   exit
;
exit
;
feature syslog
; -- SYSLOG client configuration --
   enable
   server 172.24.51.47
exit
```

You also configure events to be sent to a syslog server each time a command is executed, thus giving you a record of all the commands executed by each user. The IP of the device that sends the event and the complete command execution prompt are included in the event. As you can see below, the event also includes the user who executes the command and the remote IP address or local console he connected from.

```
06-23-2006     10:54:48       Local7.Info    172.24.78.156  Jun 23 09:46:43
CNSL:001 usr voip (172.24.79.34:209) exe *logout
06-23-2006     10:54:31       Local7.Info    172.24.78.156  Jun 23 09:46:26
CNSL:001 usr voip (172.24.79.34:209) exe Config$<Esc>
06-23-2006     10:54:27       Local7.Info    172.24.78.156  Jun 23 09:46:23
CNSL:003 usr voip (172.24.79.34:209) run Telephony Config$exit
06-23-2006     10:54:21       Local7.Info    172.24.78.156  Jun 23 09:46:16
CNSL:001 usr voip (172.24.79.34:209) exe Telephony Config$sho conf
06-23-2006     10:54:18       Local7.Info    172.24.78.156  Jun 23 09:46:13
CNSL:003 usr voip (172.24.79.34:209) run Config$telephony
06-23-2006     10:54:02       Local7.Info    172.24.78.156  Jun 23 09:45:57
CNSL:001 usr voip (172.24.79.34:209) exe *p 5
```

2.4.28 PROTOCOL

Accesses a protocol's configuration environment. You enter the desired protocol configuration by typing the **protocol** name after the command. The number of protocols available will depend on the device you have and its application license.

To access a protocol's configuration environment:

(1) Type **protocol ?** to view the list of configurable protocols:
Example:

```
Config>protocol ?
  arp      Access ARP protocol
  asrt     Access ASRT protocol
  bfd      Access BFD protocol
  bgp      Access BGP protocol
  dhcp     Access DHCP protocol
  dhcpv6  Access DHCPv6 protocol
  dls      Access DLS protocol
  dot1x    Access 802.1X protocol
  gw104    Access GW-104 protocol
  h323     Access H323 protocol
  igmp     Access IGMP protocol
  ip       Access IP protocol
  ipv6     Access IPv6 protocol
  l2tp     Access L2TP protocol
  mgcp     Access MGCP protocol
  msdp      Access MSDP protocol
  nhrp     Access NHRP protocol
  noe      Access NOE protocol
  ospf     Access OSPF protocol
  ospfv3    Access OSPFv3 protocol
  pim      Access PIM protocol
  rip      Access RIP protocol
  ripng    Access RIPNG protocol
  sccp     Access SCCP protocol
  sip      Access SIP protocol
  snmp     Access SNMP protocol
Config>protocol
```

(2) Type **protocol** followed by the name of the protocol to be configured. The prompt of the specified protocol appears. From this prompt, you can enter configuration commands that are specific to that protocol.

Example:

```
Config>protocol arp
-- ARP user configuration --
ARP config>
```

(3) Type **exit** to return to the *Config>* prompt menu.

Example:

```
ARP config>exit
Config>
```

Syntax:

```
Config>protocol <name>
```

- **<name>** is the name of the protocol whose configuration menu we want to access.

Example:

```
Config>protocol ip
-- Internet protocol user configuration --
IP config>
```

2.4.29 QUICK CONFIGURATION

Accesses the Quick Menu setup environment. This command (or functionality) is not available on all models.

For more information, please see the following manual: *Teldat Dm211-I Quick Menu*

Example:

```
Config>quick-configuration
-- Quick Configuration Menu --
Quick config>
```

Command history:

Release	Modification
11.00.02	This command is obsolete as of version 11.00.02. Quick Menu setup is no longer supported.

2.4.30 RUSH-ENGINE

Configures **Rush-engine** parameters.

Rush-engine is a traffic flow accelerator that optimizes routing performance by learning existing flows and using a cache for packets received from each flow.

When existing traffic is used to learn a flow, the latter is stored in the cache while the traffic persists. After a predefined time of traffic inactivity (idle flow timeout), the flow is removed from the cache.

Syntax:

```
Config>rush-engine ?
  disable    Disable Rush Engine
  timeout    Rush Engine idle flows timeout in seconds
  <cr>
Config>
```

- **disable** disables **Rush engine**.
- **timeout** sets the inactivity timeout period (in seconds) for established flows. Default is **5**. Valid values range from 1 to 3600 seconds.

Command history:

Release	Modification
11.01.00	This command was introduced as of version 11.01.00.

2.4.31 SAVE

Saves the configuration to the active storage unit. The active storage unit is configured using the **config-media** command.

Before the configuration is saved, and as long as the command has no accompanying parameters that indicate otherwise, the device asks the user to confirm the operation. If the user confirms the operation, the device selects the chosen media and then informs the user of the result of the operation. If the active unit comprises two media types, the configuration is saved to both and this is then indicated in the final message. Please refer to the **config-media** command in this manual and your device's installation manual for more information.

If the configuration confirmation requirement has been enabled, the configuration is saved in a temporary file (**TEMP.CFG**) awaiting confirmation. If it is confirmed (**confirm-cfg**), it is saved again, but this time under the corresponding name. If the **save** command is used in a test configuration, the configuration is saved to the **TEMP.CFG** file again without modifying the old configuration (that will be restored if the temporary configuration is not confirmed). The timer is not cleared, so once the test time has elapsed, if the device has not been restarted, the old configuration is restored. For more information, see the **confirm-cfg** and **confirm-cfg-needed** commands.

When using the **save** command, it is very important to keep in mind what configuration you are saving. There are two configurations on the device at all times. These match when the device powers on and continue to match until the user modifies one of them. One of the configurations is accessible from the RUNNING-CONFIG (Config$) process and is the one that is being used at all times. The other configuration is accessible from the CONFIG (Config>) process and is for editing purposes only.

So, if we run the **save** command from the RUNNING-CONFIG process, we will save the configuration that was being used at the time of saving. However, by using the **save** command in the CONFIG process, we will save the configuration that we edited in that process. Thus, you have to be extremely careful when modifying and saving configurations, especially when there are multiple users managing the device.

Syntax:

```
Config>save [yes [<file-name>]]
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.
- **< file-name >** is the the name of the file in which to save the configuration. If no name is entered, the device will use the active configuration filename (see **set file-cfg**).

Example 1:

```
Config>save
Save configuration (Yes/No)? y
Building configuration as text... OK
Writing configuration... OK on Flash
Config>
```

Example 2:

```
Config>save yes sample
Building configuration as text... OK
Writing configuration... OK on Flash as sample
Config>
```

2.4.32 SET

Allows you to configure some general system parameters.

Syntax:

```
Config>set <parameter>
  application-active    Permits you to select the code used to boot the router
  console               Set console configuration
  contact-person         Assign a name or identification to the contact-person
  data-link              Type of data link for a WAN line
  default-conf           Restores the default configuration
  file-cfg               Configure a configuration file as active
  ftp                    Permits you to access the FTP configuration menu
  host-location          Physical location of the router
  hostname               Assign a name to a device
  inactivity-timer       Configure the maximum inactivity time
  login                  Configure login options
  low-power-timer        Configure the time to switch to low power mode
  password               Configure the device access password
  pool                   Number of bytes assigned to each memory pool
  schedule-restart       Allows you to configure router reset schedule
  telnet                 Access the TELNET protocol configuration
  web-probe              Access the Web probe configuration
```

- **< parameter >** is the name of the parameter to configure.

Command History:

Release	Modification
11.00.06	The *pool* option is obsolete.
11.01.01	The *pool* option is obsolete.
11.01.07	The *low-power-timer* option was introduced as of version 11.01.07.

2.4.32.1 SET APPLICATION-ACTIVE

Allows you to choose which code the router will boot up with.

Syntax:

```
Config>set application-active [<code file>]
```

- **< code file >** is the name of the code file we want the device to use at boot up. If this field is left empty, the available code files are displayed.

Example:

```
Config>set application-active appcode1.bin
Config>
```

2.4.32.2 SET CONSOLE

Allows you to access the local access console configuration menu.

Syntax:

```
Config>set console
```

Example:

```
Config>set console
-- Console configuration --
Con config>?
  accounting        Set accounting options
  authorization     Set authorization options
  function          Set CONF port functionality
  login             Set login options
  speed             Set console serial port speed
  exit
Con config>
```

The commands available in the console menu are described below.

2.4.32.2.1 ACCOUNTING

Associates an accounting method list configured through the AAA facility. This way, the console service applies the methods from the accounting exec list when it registers a Shell access, and the methods from the accounting commands list when it registers an executed command.

Syntax:

Con config>accounting {commands <level> | exec} <listname>

- **commands** indicates that the method list type is accounting commands.
- **< level >** indicates the access level of the commands to be logged.
- **exec** indicates that the method list type is accounting exec.
- **< listname >** is the identifier of the accounting method list.

Example 1:

```
Con config>accounting commands 10 AccCmds
Con config>
```

In example 1, the AccCmds method list is configured to be used when accounting for a level 10 command.

Example 2:

```
Con config>accounting exec AccExec
Con config>
```

In example 2, the AccExec method list is configured to be used when accounting for a Shell access.

Method lists can only be applied if the AAA facility is enabled. Therefore, once the AAA configuration is complete, it must be enabled in order to apply the lists to the different services. Information on how to set the AAA facility up can be found in the following manual: *Teldat Dm800-I AAA Feature* .

2.4.32.2.2 AUTHORIZATION

Associates an authorization method list configured through the AAA facility. This way, the console service applies the methods from the authorization exec list when it requires Shell authorization and from the authorization commands list when it requires command authorization.

Syntax:

```
Con config>authorization {commands <level> | exec} <listname>
```

- **commands** indicates that the method list is authorization commands.
- **< level >** indicates the access level for the commands requiring authorization.
- **exec** indicates the method list is authorization exec.
- **< listname >** this is the authorization method list identifier.

Example 1:

```
Con config>authorization commands 10 AuthorCmds
Con config>
```

Example 1 specifies that the AccCmds method list will be used for authorizing level 10 commands.

Example 2:

```
Con config> authorization exec AuthorExec
Con config>
```

Example 2 specifies that the AccExec method list will be used for Shell authorization.

You can only apply method lists if the AAA facility is enabled. Therefore, once the AAA configuration is complete, it must be enabled in order to apply the lists to the different services. Information on how to set the AAA facility up can be found in the following manual: *Teldat Dm800-I AAA Feature* .

2.4.32.2.3 FUNCTION

Allows you to configure the behavior of the CONF port and display its configuration status.

Some device versions allow you to select the functionality of the local console port (CONF). In these versions, this port can behave as a local console or as an asynchronous serial port (UART).

When configuring asynchronous serial port mode, the CONF connector appears as a UART interface in the router's device list.

Example:

Config>list dev

```
Config>list dev
```

```
Interface             Connector     Type of interface
ethernet0/0           EXP/SWITCH    Marvell Fast Ethernet Switch
serial0/0             SERIAL0/WAN1  Auto Install Interface
uart0/0               CONF          Asynchronous Serial Line
x25-node              ---           Router->Node
cellular1/0           SLOT1         AT COM
cellular1/1           SLOT1         AT COM
ppp1                  ---           Generic PPP
Config>
```

This command allows the following options:

```
Con config>function ?
  set     Set CONF port functionality
  list    List CONF port functionality
Con config>
```

As this is a functionality that affects the device's BIOS (in asynchronous serial port mode during boot up, the device doesn't send data through this interface), running the **save** command does not affect this command. On the other hand, the device must be restarted for the configured value to take effect.

Note

The **save** command does not affect this command.

Note

The configured mode will not take effect until you restart the device.

Note

This command is not displayed when you run the **show configuration** command. As this is a special command that affects BIOS boot behavior, it is understood that the CONF port functionality will be configured during device installation and that it will not change later.

FUNCTION SET

Configures the operation mode of the CONF port. There are two permitted options:**console** and **asynchronous serial line**.

Syntax:

```
Con config>function set <mode>
```

- **< mode >** is the operation mode.

Example:

```
Con config>function set ?
  console              Console CLI
  async-serial-line    asynchronous serial line
Con config>function set async-serial-line
```

FUNCTION LIST

This command shows the operation mode of the CONF port.

Syntax:

```
Con config>function list
```

Example:

```
Con config>function list
 CONF port functionality: async-serial-line
Con config>
```

2.4.32.2.4 LOGIN ATTEMPTS

Allows you to configure the number of failed login attempts before blocking local console access.

Syntax:

```
Con config>login attempts <max_attempts>
```

- **< max_attempts >** is the maximum number of login attempts.

Example:

```
Con config>login attempts 2
Con config>
```

2.4.32.2.5 LOGIN AUTHENTICATION

Associates an authentication method list configured through the AAA facility. This way, the console service applies the methods of the associated list when authentication is needed.

Syntax:

```
Con config>login authentication <listname>
```

- **<listname>** is the identifier of the authentication method list.

Example:

```
Con config>login authentication AutheLogin
Con config>
```

Example 1 specifies that the AutheLogin method list be used when authentication is required for a user accessing by console.

You can only apply method lists if the AAA facility is enabled. Therefore, once the AAA configuration is complete, it must be enabled in order to apply the lists to the different services. Information on how to set the AAA facility up can be found in the following manual: *Teldat Dm800-I AAA Feature* .

2.4.32.2.6 LOGIN BLOCKING

Allows you to configure the period of time the local console will remain locked if the configured number of failed login attempts is reached.

Syntax:

```
Con config>login blocking <blocking_time>
```

- **< blocking_time >** is the time the local console will remain locked in the event that the configured number of failed login attempts is reached.

Example:

```
Con config>login blocking 1m
Con config>
```

2.4.32.2.7 SPEED

Allows you to set the **speed** (baud rate) of the local console port.

Syntax:

```
Con config>speed <baud>
```

- **< baud >** is the baud rate, in bits per second, of the local console port. Only certain values are supported.

Example:

```
Con config>speed ?
  9600      bits per second
  14400     bits per second
  19200     bits per second
  38400     bits per second
  57600     bits per second
  115200    bits per second
Con config>speed 115200
Con config>
```

2.4.32.2.8 EXIT

Returns to the previous prompt.

Syntax:

```
Con config>exit
```

Example:

```
Con config>exit
Config>
```

2.4.32.3 SET CONTACT-PERSON

Allows you to provide a contact person name/ID for this router. The name can have a maximum of 79 characters. You can view this information by typing **list configuration**.

Syntax:

```
Config>set contact-person <name>
```

- **< name >** is the name or identification of the contact person.

Example:

```
Config>set contact-person Antonio Leon
Config>
```

2.4.32.4 SET DATA-LINK

Sets the **data link** type to use for a WAN line.

Syntax:

```
Config>set data-link <type> <interface name>
```

- **< type >** is the type of data link that to apply to the WAN line.

To find out what types are available, use the **set data-link ?** command.

Example:

```
Config>set data-link ?
  arly           Alarm Relay on asynchronous data link for a WAN line
  asdp           ASDP data link for a WAN line
  astm           ASTM data link for a WAN line
  async          Asynchronous data link for a WAN line
  at             AT modem data link for a WAN line
  frame-relay    Frame-Relay data link for a WAN line
  scada          SCADA data link for a WAN line
  sdlc           SDLC data link for a WAN line
  sepi           SEPI data link for a WAN line
  sync           Synchronous data link for a WAN line
  udafo          Udafo data link for a WAN line
  x25            X25 data link for a WAN line
  x28            X28 data link for a WAN line
```

- **< interface name >** is the name of the WAN interface on which we want to apply the specified data link type.

To find out what WAN interfaces are available on the device, type the **list devices** command.

Example:

```
Config>list devices
Interface            Connector     Type of interface
ethernet0/0          GE0/FE0/LAN1  Fast Ethernet interface
ethernet0/1          GE1/FE1/LAN2  Fast Ethernet interface
serial0/0            SERIAL0/WAN1  Auto Install Interface
serial0/1            SERIAL1/WAN2  X25
bri0/0               BRI/ISDN1     ISDN Basic Rate Int
```

```
x25-node             ---           Router->Node
Config>
```

Example:

```
Config>set data-link frame-relay serial0/0
Config>list devices
Interface            Connector     Type of interface
ethernet0/0          GE0/FE0/LAN1  Fast Ethernet interface
ethernet0/1          GE1/FE1/LAN2  Fast Ethernet interface
serial0/0            SERIAL0/WAN1  Frame Relay
serial0/1            SERIAL1/WAN2  X25
bri0/0               BRI/ISDN1     ISDN Basic Rate Int
x25-node             ---           Router->Node
Config>
```

You can then check if the command has succeeded by typing the **list devices** command.

2.4.32.5 SET DEFAULT-CONFIG

Deletes the current configuration and restores the default configuration. This command (or functionality) is not available on all models.

Syntax:

```
Config>set default-conf [yes]
```

Example:

```
Config>set default-conf yes
Config>
```

2.4.32.6 SET FILE CFG

Allows you to select the active configuration file. This is the file that will be processed when the device restarts.

It also shows the active storage unit. For more information on how to change the active storage unit, please see the **config-media** command in this manual.

Syntax:

```
Config>set file-cfg [<file name>]
```

- **< file name >** is the name of the configuration file to be activated.

The file name is indicated without extension. If none are passed by the command line, the device lists all the *cfg* files available. If the selected file does not exist, the device will use the default configuration at startup.

Example:

```
Config>set file-cfg
Config Media: Flash only
 A:                       ROUTER        494   12/05/06   20:15   Flash
 A:                       SAMPLE        523   12/11/06   15:15   Flash
Current config: ROUTER
Config>set file-cfg sample
Config>
```

2.4.32.7 SET FTP

Accesses the *FTP* (File Transfer Protocol) configuration menu. See the associated manual: *Teldat Dm724-I FTP/sFTP Protocol.*

Syntax:

```
Config>set ftp
```

Example:

```
Config>set ftp
-- FTP user configuration --
FTP config>
```

2.4.32.8 SET HOST-LOCATION

Allows you to enter the physical location of the router. You can view this information by typing **list configuration**.

Syntax:

```
Config>set host-location <place>
```

- **< place >** is the location of the device. The location can have a maximum of 79 characters.

Example:

```
Config>set host-location Tres cantos (Madrid)
Config>
```

2.4.32.9 SET HOSTNAME

Allows you to assign a device name to the router. You can view this information by typing **list configuration**.

Syntax:

```
Config>set hostname <name>
```

- **< name >** is the name of the device. The name can have a maximum of 79 characters.

Example:

```
Config>set hostname SuperRouter
Config>
```

2.4.32.10 SET IGNITION-OFF-POWERDOWN-TIMER

Note

This command is only available on devices that have the power management functionality; usually those intended for use in vehicles and powered by the vehicle's own battery. Please consult your device's installation manual to see whether your router supports this functionality.

Allows you to configure the length of time the device will remain on after the ignition is turned off in the vehicle where the device is installed. The goal is for the device to automatically shut down after a configured period of time to save the vehicle battery.

Syntax:

```
Config>set ignition-off-powerdown-timer [<time> | disabled]
```

- **< time >** is the time (in minutes) that the device stays on after the ignition is turned off in the vehicle where the device is installed. After the timer expires, the device performs an orderly shutdown and then switches off. Valid values range from 0 to 1044 minutes (24 hours).
- **disabled** disables the timer so that the device shuts down as soon as the vehicle ignition is turned off.

Example:

```
Config>set ignition-off-powerdown-timer 20
Config>
```

- This functionality is disabled by default. The device shuts down as soon as the vehicle ignition is turned off (equivalent to set ignition-off-powerdown-timer disabled) if you do not set a time interval.

2.4.32.11 SET INACTIVITY-TIMER

Allows you to set a maximum inactivity time for remote terminal connections (TELNET). If the maximum inactivity time is reached, the device's Telnet server is disconnected.

This maximum inactivity time also applies to the device's local console connection. If the configured inactivity time expires before any keys have been pressed, the local connection is closed and the user must re-enter the password to be able to use the console again.

Syntax:

```
Config>set inactivity-timer [<time> | disabled]
```

- **< time >** is the maximum inactivity time, in minutes, before the telnet connection is closed or the console locked. Valid values range from 1 minute to 10 hours.
- **disabled** disables the timer so that the telnet connection is not closed and the console is not locked, unless the remote device closes the telnet connection or the user terminates the session using the **logout** command.

Example:

```
Config>set inactivity-timer 20
Config>
```

The inactivity timer is set to 10 minutes (equivalent to set inactivity-timer 10) by default.

2.4.32.12 SET LOGIN

Allows you to configure parameters related to the device access name.

Syntax:

```
Config>set login <option>
  case-sensitive    Activate login case-sensitiveness
```

- **< option >** is the option to configure.

2.4.32.12.1 SET LOGIN CASE-SENSITIVE

Enables case-sensitivity in the username authentication process for console, telnet and ftp access.

Syntax:

```
Config>set login case-sensitive
```

Example:

```
Config>set login case-sensitive
Config>
```

By default, case sensitivity in the username authentication process is disabled on the device.

2.4.32.13 SET LOW-POWER-TIMER

Allows you to configure the length of time the device will wait before switching to low power mode.

The manner in which to quit the low power mode depends on the device. Typically, the device will resume operation when a WoL packet arrives to the Ethernet port used in BIOS or when the factory reset button is pressed.

Syntax:

```
Config>set low-power-timer [<time> | disabled]
```

- **< time >** is the time (in minutes) that the device waits before switching to low power mode. Valid values range from 1 to 20 minutes.
- **"disabled"** disables the timer, meaning the device won't ever enter into low power mode.

Example:

```
Config>set low-power-timer 20
Config>
```

- This entry will set the device in low power mode after 20 minutes if none of its interfaces has sent or received traffic.

This functionality is disabled by default.

> **Note**
>
> This command is only available for devices that support the low power functionality. This feature allows the device to switch to low power when no traffic has been sent or received in a while, in order to save energy and meet the requirements of the ErP directive.

Command History:

Release	Modification
11.01.07	The *low-power-timer* option was introduced as of version 11.01.07.

2.4.32.14 SET PASSWORD

Allows you to use a local console connection, a remote Telnet connection, or an FTP connection to configure the device's access password.

Syntax:

```
Config>set password <word>
```

- **< word >** is the password to set.

Example:

```
Config>set password mk34po99
Config>
```

2.4.32.15 SET POOL

Allows you to configure memory allocation in the device's memory POOLs. A bad setup can leave your device unusable.

Syntax:

```
Config>set pool <option> <size>
  iorbs    Iorbs pool size in bytes
  msg      Message pool size in bytes
```

- **< option >** is the name of the pool whose size, in bytes, we want to set.
- **< size>** is the size, in bytes, that we want to give the pool.

Running the **set pool ?** command displays a list of all the pools available for configuration.

Example:

```
Config>set pool iorbs 2048000
Config>
```

Note

This command is dangerous and must only be used by qualified personnel. A bad setup can leave your device unusable.

Command history:

Release	Modification
11.00.06	This command is obsolete.
11.01.01	This command is obsolete.

2.4.32.16 SET SCHEDULE-RESTART

This command is used when you want the device to perform a periodic restart at a specific time or a certain amount of time after start up. This is useful for devices that are difficult to access and that, due to the installation conditions, are likely to lose connectivity. In many cases, a device restart can restore connectivity.

Syntax:

```
Config> set schedule-restart time <day-time> offset <time-offset>
  time      Set time of day at which device will restart
  offset    Set Offset for periods longer than 24 hours
```

- <**day-time**> is the time of day you want the device to restart or, if *time-offset* is configured with a value other than 0, the time of day the countdown indicated in *time-offset* begins. It must be expressed in hh:mm format, except when the value is 0.
- **< time-offset>** indicates the time offset value for restarting the device. Its units are seconds, although you can enter directly: weeks (w), days (d), hours (h), minutes (m) and seconds (s) (e.g., 1w2d3h). The *day-time* value is used differently depending on the value you configure:

  - day-time = 0. Indicates the period of time that must elapse after device startup before performing a device restart.

- day-time # 0. Indicates the period of time that must elapse from the time indicated in *day-time* before performing a device restart.

Example:

```
Config> set schedule-restart time 03:30 offset 2d
Config>
```

The command allows you to configure three different types of scheduled restart:

- Every day at the same time:

  To have this restart, the offset value must not be set or it must be given a value of 0.

  In the following example, the router restarts every day at **03:30**.

  ```
  Config> Config>set schedule-restart time 03:30
  Config>
  ```

  Or:

  ```
  Config> Config>set schedule-restart time 03:30 offset 0
  Config>
  ```

- Every few days or one hour + offset:

  You need to program both the restart time and the desired period.

  The logical thing is to program the offset as a multiple of one day; otherwise you would have to calculate when the device reset is going to take place (time + offset).

  In any case, the device has a 180-second time window so that, if at the time of startup the current time is between the set time + 180 seconds, it understands that it only has to check the *offset* field, since it is a periodic restart at the same time. This needs to be taken into account when setting an offset value of less than one day.

  In this example it will restart every 2 days at 16:04.

  ```
  Config> Config> set schedule-restart time 16:04 offset 2d
  Config>
  ```

- A period of time after startup:

  For this restart, you need to set the *time* field to 0. In the offset field, you enter the time that must elapse from starting the device until the restart takes place.

  In this example it will restart every 18 hours.

  ```
  Config> Config> set schedule-restart time 0 offset 18h
  Config>
  ```

We do not recommend the first method in devices without a battery-powered real-time clock or some other method of updating the system clock (NTP).

To delete the programming, you can set both fields to **0**, or use the **no** command. For example:

```
Config> Config>no set schedule-restart time 16:04 offset 2d
Config>
```

⚠ **Important**

This command may compromise device integrity. Before loading a new software version, firmware module or saving settings, you must ensure that the automatic process DOES NOT restart the device.

2.4.32.17 SET TELNET

Accesses the *TELNET* protocol configuration. For more information, please see the following manual: *Teldat Dm738-I TELNET Protocol*.

Syntax:

```
Config>set telnet
```

Example:

```
Config>set telnet
-- Telnet user configuration --
Telnet config>
```

2.4.32.18 SET WEB-PROBE

Accesses the *Web probe* configuration environment. This command (or functionality) is not available on all models.

Syntax:

```
Config>set web-probe
```

Example:

```
Config>set web-probe
-- Web Probe user configuration --
Probe config>
```

For more information about the *Web probe* configuration environment, please see the following manual: *Teldat Dm749-I NSM (Network Service Monitor)* .

2.4.33 STRONG PASSWORD

Enables user password checking while preventing the use of weak passwords. The password strength level can be a value from 0 to 5, with 0 being the default value and allowing you to configure the strongest password.

Syntax:

```
Config>strong-password ?
  <0..5>    Level of strength (0: strongest)
  <cr>
```

Example:

```
Config>strong-password 1
```

2.4.34 TELEPHONY

Accesses the Voice over IP (VoIP) configuration environment.

Syntax:

```
Config>telephony
```

Example:

```
Config>telephony
-- Telephony configuration --
Telephony Config>
```

For more information, please see the following manual: *Teldat Dm722-I Telephony Over IP.*

2.4.35 TIME

Allows you to change and view the device's date and time settings. It is also used to configure the start and end of summer time (or daylight saving time).

Syntax:

```
Config>time <option> [parameters]
  list            Check the date and time of the device
  no              Negates a command or sets its defaults
  set             Change the date and time of the device
  summer-time     Configure summer (daylight savings) time
  timezone        Changes the difference in hours with respect to UTC times
```

- **< option >** is the name of the action to be performed.
- **[parameters]** are the parameters required for the specified option.

The available options are:

2.4.35.1 TIME LIST

Allows you to check the device's date and time.

Syntax:

```
Config>time list
```

Example:

```
Config>time list
Set by: operator
Date: Wednesday, 03/02/05          Time: 16:28:46 CET
Config>
```

2.4.35.2 TIME SET

Allows you to change the device's date and time settings.

Syntax:

```
Config>time set <month> <day> <year> <week day> <hour> <minute> <seconds>
```

- **< month >** is the month to set.
- **< day >** is the day to set.
- **< year >** is the year to set.
- **< week day >** is the day of the week to set.
- **< hour >** is the hour to set.
- **< minute >** are the minutes to set.
- **< seconds >** are the seconds to set.

Example:

```
Config>time set 3 2 5 3 16 29 59
Config>
```

Note

Some devices lose the date and time settings when they are restarted. In such cases, you will need to configure these parameters using the NTP protocol. For more information, please see the following manual: *Teldat Dm728-I NTP Protocol* .

2.4.35.3 TIME NO

Negates a command within the time configuration environment or sets its defaults.

Syntax:

```
Config>time no <option>
```

- **< option >** specifies the selected option. The available options are **summer-time** and **timezone**.

2.4.35.3.1 *TIME NO SUMMER-TIME*

Clears the previously configured summer time application period.

Syntax:

```
Config>time no summer-time <option>
 zone-name     Deletes summer time zone name
 <cr>          Deletes summer time configuration
```

- **< option >** specifies the selected option.

i) time no summer-time zone-name

Deletes only the summer-time time zone.

Syntax:

```
Config>time no summer-time zone-name
```

Example:

```
Config>sho menu
; Showing Menu Configuration for access-level 15 ...
log-command-errors
no configuration
set inactivity-timer disabled
time summer-time recurring 1 mon jan 03:00 1 mon dec 03:00
time summer-time zone-name "CET"
;
dump-command-errors
end
Config>time no summer-time zone-name
Config>show menu
; Showing Menu Configuration for access-level 15 ...
log-command-errors
no configuration
set inactivity-timer disabled
time summer-time recurring 1 mon jan 03:00 1 mon dec 03:00
;
dump-command-errors
end
Config>
```

ii) time no summer-time

Deletes all the summer time settings, including the start and end dates and times and the time zone.

Syntax:

```
Config>time no summer-time
```

Example:

```
Config>show menu
; Showing Menu Configuration for access-level 15 ...
log-command-errors
no configuration
set inactivity-timer disabled
time summer-time recurring 1 mon jan 03:00 1 mon dec 03:00
time summer-time zone-name "CET"
;
dump-command-errors
end
Config>time no summer-time
Config>show menu
; Showing Menu Configuration for access-level 15 ...

log-command-errors
no configuration
set inactivity-timer disabled
dump-command-errors
end
Config>
```

2.4.35.4 TIME SUMMER-TIME

Configures when summer time is in effect for the device and the one-hour offset which must be added to the clock value that would be obtained if it were not within that period.

```
Config>time summer-time <option>
  date         Configure absolute summer time
  list         Display configured summer time
  recurring    Configure recurring summer time
  zone-name    Configure summer time zone name
```

- **< option >** specifies the selected option.

There are four options available (three of them configuration and one monitoring).

2.4.35.4.1 *TIME SUMMER-TIME DATE*

Allows you to configure when summer time is in effect using exact dates and times. Here you are configuring absolute summer time.

Syntax:

```
Config>time summer-time date <start day> <start
                        month> <start
                        year> <start
                        hour> <end
                        day> <end
                        month> <end
                        year> <end
                        hour>
```

- **< start day >** specifies the day of the month to start summer time.
- **< start month >** specifies the month to start summer time. Use the first three letters of the month (jan/feb/mar/apr/may/jun/jul/aug/sep/oct/nov/dec) to specify the month.
- **< start year >** specifies the year to start summer time. Use the last two digits of the year to specify the year, e.g., 05 for 2005.
- **< start hour >** specifies the time (HH:MM) to start summer time.
- **< end day >** specifies the day of the month to end summer time.
- **< end month >** specifies the month to end summer time. Use the first three letters of the month (jan/feb/mar/apr/may/jun/jul/aug/sep/oct/nov/dec) to specify the month.
- **< end year >** specifies the year to end summer time. Use the last two digits of the year to specify the year, e.g., 05 for 2005.
- **< end hour >** specifies the time (HH:MM) to end summer time.

Example:

```
Config>time summer-time date 30 mar 05 02:00 26 oct 05 03:00
Config>
```

2.4.35.4.2 *TIME SUMMER-TIME RECURRING*

Allows you to configure when summer time is in effect by setting the beginning and end of the period in a relative way using the day of the week, week number, month, hour, and minute from which to consider the one-hour offset, and the same parameters to configure the return to standard time. Once these data are entered, the device will automatically switch to summer time and revert back to standard time each year.

Syntax:

```
Config>time summer-time recurring <start
                        week number> <start
                        week
                        day> <start
                        month> <start
                        hour> <end
                        week number> <end
                        week
                        day> <end
                        month> <end
                        hour>
```

- **< start week number >** specifies the week of the month (1 to 5) to start summer time.
- **< start week day >** specifies the day of the month to start summer time. Use the first three letters of the weekday (**mon/tue/wed/thu/fri/sat/sun**) to specify the day of the week.
- **< start month >** specifies the month to start summer time. Use the first three letters of the month (**jan/feb/mar/apr/may/jun/jul/aug/sep/oct/nov/dec**) to specify the month.
- **< start hour >** specifies the time, in 24 hour format (HH:MM), to start summer time.
- **< end week number >** specifies the week of the month (1 to 5) to end summer time.
- **< end week day >** specifies the day of the month to end summer time. Use the first three letters of the weekday (**mon/tue/wed/thu/fri/sat/sun**) to specify the day of the week.
- **< end month >** specifies the month to end summer time. Use the first three letters of the month

(jan/feb/mar/apr/may/jun/jul/aug/sep /oct/nov/dec) to specify the month.

- **< end hour >** specifies the time, in 24 hour format (HH:MM), to end summer time.

Example:

If summer time were to start on the fourth (*start week number* = 4) Sunday (*start week day* = sun) of March (*start month* = mar) at 2 a.m. (*start hour* = 02:00), and end on the fourth (*end week number* = 4) Sunday (*end week day* = sun) in October (*end month* = oct) at 3 a.m. (*end hour* = 03:00), you would enter the following:

```
Config>time summer-time recurring 4 sun mar 02:00 4 sun oct 03:00
Config>
```

Sometimes, however, this rule may not be flexible enough. Note, for example, that there are four Sundays in March 2007, while the same month in 2008 has five. To allow for this, set the week number in the month to 5 (using the **< startweek number >** and/or **< endweek number>** parameters). Because of the special significance that these parameters have when they are set to 5, the changes are applied on the last weekday configured within the configured month, regardless of whether it is a five- or only a four-week month.

Example:

If we want summer time to start each year on the last Sunday in March at 2 a.m. and finish each year on the last Sunday in October at 3 a.m., we must type:

```
Config>time summer-time recurring 5 sun mar 02:00 5 sun oct 03:00
Config>
```

2.4.35.4.3 *TIME SUMMER-TIME ZONE-NAME*

Configures the name of the summer-time time zone.

The assigned text can be up to 15 characters long.

Syntax:

```
Config>time summer-time zone-name <text>
```

- **< text >** specifies the summer-time time zone name. If the text contains spaces, it needs to be be placed between quotation marks.

Example:

```
Config>time summer-time zone-name CET
Config>
```

The following table includes acronyms commonly used in different time zones for this parameter.

ACRONYM	TIME ZONE NAME AND OFFSET WITH RESPECT TO UTC
Europe	
GMT	Greenwich Mean Time, as UTC
BST	British Summer Time, as UTC + 1 hour
IST	Irish Summer Time, as UTC + 1 hour
WET	Western Europe Time, as UTC
WEST	Western Europe Summer Time, as UTC + 1 hour
CET	Central Europe Time, as UTC + 1
CEST	Central Europe Summer Time, as UTC + 2
EET	Eastern Europe Time, as UTC + 2
EEST	Eastern Europe Summer Time, as UTC + 3
MSK	Moscow Time, as UTC + 3
MSD	Moscow Summer Time, as UTC + 4
United States and Canada	
AST	Atlantic Standard Time, as UTC –4 hours
ADT	Atlantic Daylight Time, as UTC –3 hours
ET	Eastern Time, either as EST or EDT, depending on place and time of year
EST	Eastern Standard Time, as UTC –5 hours
EDT	Eastern Daylight Saving Time, as UTC –4 hours
CT	Central Time, either as CST or CDT, depending on place and time of year

CST	Central Standard Time, as UTC –6 hours
CDT	Central Daylight Saving Time, as UTC –5 hours
MT	Mountain Time, either as MST or MDT, depending on place and time of year
MST	Mountain Standard Time, as UTC –7 hours
MDT	Mountain Daylight Saving Time, as UTC –6 hours
PT	Pacific Time, either as PST or PDT, depending on place and time of year
PST	Pacific Standard Time, as UTC –8 hours
PDT	Pacific Daylight Saving Time, as UTC –7 hours
AKST	Alaska Standard Time, as UTC –9 hours
AKDT	Alaska Standard Daylight Saving Time, as UTC –8 hours
HST	Hawaiian Standard Time, as UTC –10 hours
Australia	
WST	Western Standard Time, as UTC + 8 hours
CST	Central Standard Time, as UTC + 9.5 hours
EST	Eastern Standard/Summer Time, as UTC + 10 hours (+11 hours during summer time)

2.4.35.4.4 *TIME SUMMER-TIME LIST*

Displays the summer time settings.

Syntax:

```
Config>time summer-time list
```

Example 1:

```
Config>time summer-time list
Absolute summer time configured:
Start: Date: 30/03/03   Time: 02:00
End:   Date: 26/10/03   Time: 03:00
Zone-name: CET
Config>
```

Example 2:

```
Config>time summer-time list
Recurring summer time configured:
Start: Sunday 4th week of March at 02:00
End:   Sunday 4th week of October at 03:00
Zone-name: CET
Config>
```

Finally, the **show config** command provides the summer-time settings in text mode.

Example 1:

```
Config>show menu
; Showing Menu Configuration for access-level 15 ...
log-command-errors
no configuration
time summer-time date 30 mar 3 02:00 26 oct 3 03:00
;
dump-command-errors
end
Config>
```

Example 2:

```
Config>show menu
; Showing Menu Configuration for access-level 15 ...
log-command-errors
no configuration
time summer-time recurring 4 sun mar 02:00 4 sun oct 03:00
;
dump-command-errors
end
Config>
```

2.4.35.5 TIME TIMEZONE

Configures the time offset from UTC, thus determining the time zone of the device.

Syntax:

```
Config>time timezone <-12…12>
```

Example:

```
Config>time timezone 1
Config>
```

Note

Valid values range from -12 to 12. Default is 1.

2.4.36 UCI

Allows you to configure the router's encryption unit.

Syntax:

```
Config>uci <option> [parameters]
  cfg
  change cfg
  keys
  lqueue
  mode
  table
  user_password
  lqueue
```

- **< option >** specifies the selected option.
- **[parameters]** are the parameters that need to be entered depending on the selected option.

2.4.37 UNSET-DEMO-LICENCE

Deactivates the demo license on the device so that the base license is used after a reboot.

This command will only run if there is an active demo license.

Syntax:

```
Config> unset-demo-licence
```

Example:

```
Config>unset-demo-licence
Demo licence disabled. Restart device to start using the base licence
Config>
```

2.4.38 USER

Allows you to create and configure users with access permissions.

2.4.38.1 Creating a user

Syntax:

```
Config>user <name> <password | hash-password> <pwd>
```

- **< name >** is the name of the new user to create on the device.
- **<password | hash-password>** specifies the password format - in clear text or hash-code - to enter next for the specified user.
- **< pwd>** is the access password in the chosen format to give to the specified user.

Example:

```
Config>user usersample password trescantos1985
Config>
```

For security reasons, when using the **show configuration** command to view the configuration of a device on which users have been created, the command lines used for creating users will always be hashed with the hash-password option, regardless of whether or not said option was originally used to create them. This means that a user's clear text password cannot be extracted by reading the device's configuration file. If a user or administrator forgets a password, you must recreate the user with a new password.

Example:

```
Config>show menu
; Showing Menu Configuration for access-level 15 ...
log-command-errors
no configuration
user usersample hash-password E7AE08B3FEB1F020EEEDE75FCD0D41F1
;
dump-command-errors
end
Config>
```

2.4.38.2 User management

Once you have created two or more users, you can manage their access levels, enable or disable their access, and so on. Use the following process to access a user's configuration menu:[1]

Syntax:

```
Config>user <name> <option> <parameter>
  access-level     Specify the user access level
  active           Activate the user
  hash-password    Entry the hash of the password
  keymanager       IPSec keys manager
  no               Negates a command or sets its defaults
  password         Entry the password
```

- **< name >** is the name of the user to manage.
- **<option>** specifies the management operation to perform.
- **< parameter>** are the parameters required for the selected option.

Available management operations (options) include:

2.4.38.2.1 *access-level*

Specifies the user's access level.

Syntax:

```
Config>user <name> access-level <level> <mode>
  <level>
    <0..15>          value in the specified range
    configuration    Configuration access level [10]
    events           Events access level [1]
    monitor          Monitor access level [5]
    none             None access level [0]
    root             Root access level [15]
  <mode>
    strict    Restricts user access level to exactly the specified value
    <cr>
```

- **< name >** is the name of the user to manage.
- **< level >** is the desired level of access to assign to the specified user. You can choose a value between 0 and 15, either specifying the digit or using the configuration, events, monitor, none or root tags.
- **< mode >** is the device's operating mode in relation to the configured access level. There are two modes:

  - Default. The user can run commands that require an execution level that is less than or equal to his/her access level.

  - Strict. The user can run commands that require an execution level that is exactly equal to his/her access level.

[1] Performing user management before you have at least two users makes no sense: if there is only one user, that user has to be the administrator, or *root* user, and thus must have all privileges enabled.

Example:

```
Config>user usersample2 access-level 8 strict
Config>
```

You can view user access levels by using the **list user** command in the configuration console.

By default, newly created users are assigned root-level access in default mode.

2.4.38.2.2 *active*

Enables the user to access the device.

Syntax:

```
Config>user <name> active
```

- **< name >** is the name of the user to enable.

Example:

```
Config>user usersample2 active
Config>
```

You can view user access levels by using the **list user** command in the configuration console.

New users are granted access to the device by default.

2.4.38.2.3 *hash-password*

Configures the user's password hash code.

Syntax:

```
Config>user <name> hash-password <hash code>
```

- **< name >** is the name of the user to manage.
- <**hash code**> is the password hash code to assign to the specified user.

Example:

```
Config>user usersample2 hash-password E7AE08B3FEB1F020EEEDE75FCD0D41F1
Config>
```

You can view user access levels by using the **show configuration** command in the configuration console.

2.4.38.2.4 *keymanager*

Configures a user as an IPSec *keymanager* on the device. Only *root* users or another *keymanager* (if there is one) can perform this action.

This command partitions the permissions management system by creating two profiles that are completely separate: one is only used to configure IPSec keys, while the other configures the remaining device parameters.

Syntax:

```
Config>user <name> keymanager
```

- **< name >** is the name of the user to configure.

Example:

```
Config>user usersample2 keymanager
Config>
```

> **Note**
>
> When this type of special user exists, there are no longer any users with complete control of the device's configuration system. Therefore, the use of this option is not recommended except in special cases where device management is shared.

You can view the users configured as *keymanager* by using the **list user** command in the configuration console.

2.4.38.2.5 *no*

Undoes the action of a command or sets its default values.

Syntax:

```
Config>user <name> no <option>
  active        Deactivate the user
  keymanager    IPSec keys manager
```

- **< name >** is the name of the user to configure.
- **< option >** specifies the operation to be performed.

There are two possible options:

2.4.38.2.6 *active*

Disables access to the device for a certain user.

Syntax:

```
Config>user <name> no active
```

- **< name >** is the name of the user.

Example:

```
Config>user usersample2 no active
Config>
```

2.4.38.2.7 *keymanager*

If you have configured a user as IPSec *keymanager* on the device, you can use this option to undo that configuration.

Syntax:

```
Config>user <name> no keymanger
```

- **< name >** is the name of the user.

Example:

```
Config>user usersample2 no keymanager
Config>
```

2.4.38.2.8 *password*

Sets a user's password.

Syntax:

```
Config>user <name> password <text>
```

- **< name >** is the name of the user to manage.
- <**text**> is the password to assign to the specified user.

Example:

```
Config>user usersample2 password trescantos1985
Config>
```

2.4.39 END

Marks the end of a configuration file. This command must be included at the end of all configuration files to ensure they load properly during startup.

The command is automatically appended as the last command in the device's configuration files (cfg) when you run the **save** or **show config** commands.

Syntax:

```
Config>end
```

Example:

```
Config>show config
; Showing System Configuration for access-level 15 ...
; ATLAS50 Router 9 48 Version 10.7.0
log-command-errors
no configuration
;
;
network ethernet0/0
; -- Ethernet Interface User Configuration --
   no ip address
;
exit
;
;
;
network ethernet0/1
; -- Ethernet Interface User Configuration --
   no ip address
;
exit
;
;
;
;
network x25-node
; -- X25-node interface configuration --
   no ip address
;
Exit
;
;
;
dump-command-errors
end
; --- end ---
Config>
```

Chapter 3 Router monitoring

3.1 Introduction

This chapter describes the device's monitoring (p3) process. This process lets you display information about the status of the system, as well as statistics gathered by the device. It also enables users with the appropriate access rights to manage those statistics and perform functional testing on certain device functionalities.

The following command types are available in the monitoring process:

- *Event* (display/hide events -view/hide-). The user must have, at least, EVENT-level access to run these commands.
- *Show* (statistics list, interface status, counters, etc.). The user must have, at least, MONITOR-level access to run these commands. Other commands found within this typology include *telnet, ssh, ping, vrf-ping, atm-ping, traceroute*, etc.
- *Clear* (delete/reset statistical information, registers, counters, etc.). The user must have, at least, CONFIG-level access to run these commands.
- *Conf* (interface testing, event management, IPSec tunnel activation, etc.). The user must have, at least, CONFIG-level access to run these commands. Other commands found within this typology include *bping, vrf-bping, tftp*, etc.
- *Root* (send an escape character to another console terminal connected to the device to force a user to exit a menu or to terminate a session). The user must have ROOT-level access (level 15) to run these commands.

If the user requests help (?) from the monitoring process, the only commands displayed are the ones the user is allowed to execute.

The access levels available and the configuration process are outlined in *Router console* on page 3 (section *Connecting to the Teldat Router* on page 4) and *Teldat Router Configuration* on page 19 (section *USER* on page 95) herein.

3.2 Monitoring commands

Enter/exit MONITORING

To enter the monitoring process from the GESTCON Management Console prompt (*), type the **monitor** or **process** command followed by the configuration process number, which in this case is **3**.

Example:

```
*monitor
Console Operator
+
```

To exit the monitoring process and return to the GESTCON Management Console prompt (*), type the escape character (*Ctrl-p* by default).

Command	Function
? (HELP)	Displays a list of monitoring commands.
BUFFER	Displays information about the packet buffers assigned to each interface.
CLEAR	Clears network statistics.
CONFIGURATION	Lists the status of the current protocols and interfaces.
DEVICE	Displays network hardware statistics or statistics for the specified interface.
ERROR	Displays the error counters.
EVENT	Enters the event registration system environment.
FEATURE	Accesses monitoring commands for router facilities which are outside the usual protocol and network interface monitoring processes.
HARDWARE	Sets hardware configuration.
LAST-CONFIG-CHANGES	Shows the last configuration changes made.
MALLOC-MONITOR	Accesses device memory management monitoring commands.
MANAGEMENT	Enters the master router environment.
MEMORY	Displays memory, buffer and data packets.

NETWORK	Enters the specified network's console environment.
NODE	Enters the node monitoring environment.
PROTOCOL	Enters the specified network's command environment.
QUEUE	Displays the buffer statistics for a specific interface.
QUICK	Accesses quick menu monitoring.
RUSH-ENGINE	Accesses rush engine monitoring.
STATISTICS	Displays the statistics for a specific interface.
SYSTEM	Allows you to monitor system memory, batteries and CPU usage; set the console port speed; display the firmware needed for the proper functioning of the device; activate certain debugging information; display user login history; view open Telnet/SSH sessions; and exchange commands or messages between the terminals corresponding to those open sessions.
TFTP	Accesses the TFTP client on the device.
TELEPHONY	Accesses the monitoring environment of the device's telephony functions.
UCI	Encryption statistics.
UPTIME	Shows how long the router has been running since the last boot.
VERSION	Shows the software version.
VISORNET	Accesses the VisorNet monitoring environment.
WEB-PROBE	Accesses web probe monitoring.
LOG	Enables or displays the event log level for events not included in the Event Logging System.

Home command

You can use the **home** command in all of the monitoring process menus. It returns you to the monitoring process regardless of the menu or submenu you are currently on. This commands provides a way to return to the monitoring process without having to exit the menus one by one with the "exit" command.

Examples:

```
*p 3
Console Operator


+protocol DHCP


DHCP Protocol monitor

DHCP+server


DHCP-Server+home


+
```

Command history:

Release	Modification
11.01.03	This command was introduced as of version 11.01.03.

Root command

The **root** command is available in all menus that can be accessed during the configuration and monitoring processes. It allows you to return to the **root** prompt, regardless of the menu or submenu you are currently on.

Examples:

```
*
*p 3
Console Operator
```

```
+protocol IP
-- IP protocol monitor --
IP+ipsec
-- IPSec protocol monitor --
IPSec+root

*
```

Command history:

Release	Modification
11.01.06	This command was introduced as of version 11.01.06.

3.2.1 ? (HELP)

Lists available commands from the current prompt. You can also type a question mark (?) after a specific command to list its options.

Syntax:

```
+?
```

Example:

```
+?
  buffer                  Packet buffers assigned to each interface
  clear                   Clear network statistics
  configuration           List status of current protocols and interfaces
  device                  List statistics for the specified interface
  error                   List error counters
  event                   Event Logging System environment
  feature                 Access to monitoring commands for router features
  hardware                Set hardware configuration
  last-config-changes     Display the last changes made in the configuration
  log                     Dump log data
  malloc-monitor          Malloc monitor information
  management              Master router environment
  memory                  Display memory, buffer and packet data
  network                 Enter the console environment of a specified network
  node                    Enter the node monitoring environment
  protocol                Enter the commands environment for a specified
                          protocol
  queue                   Display buffer statistics for a specified interface
  quick                   Access the quick menu monitoring
  rush-engine             Rush Engine Monitor
  statistics              Display statistics for a specified interface
  system                  Permit monitoring of the system's memory and stacks
  telephony               Monitoring environment for the telephony functions
  uci                     Encryption statistics
  web-probe               Access the Web poll monitoring
  exit
+
```

3.2.2 BUFFER

Displays information about the packet buffers assigned to each interface.

Note

Each buffer on a device is of the same size and built dynamically. Buffer sizes vary from one device to another.

Syntax:

```
+buffer [<verbose> | <interface>]
```

- **< interface >** is the name of the interface about which we want to display information.
- **< verbose >** displays additional information.

To obtain the available interfaces on the device, type the **configuration** command at the plus prompt (**+**). If no name is entered, the command displays information about all of the interfaces.

Example:

```
+buffer
                  Input Buffers          Buffer       Sizes
Interface        Req  Alloc   Low  Curr  Hdr Wrap   Data Trail  Total
ethernet0/0       40     40     5   40   22   62   1500     4   1588
serial0/0         40     40     5   40   18   62   2048    12   2140
serial0/1          0      0     0    0    0   62      0     0     62
serial0/2          0      0     0    0    0   62      0     0     62
bri0/0           160    160     5  160   17   62   2048    12   2139
x25-node           0      0     0  100   20   62   1500     0   1582

Buffer size:    2144
Packet size:    2048
Trailer size:   12
Packet offset:  84

+ buffer ethernet0/0
                  Input Buffers          Buffer       Sizes
Interface        Req  Alloc   Low  Curr  Hdr Wrap   Data Trail  Total
ethernet0/0       40     40     5   40   22   62   1500     4   1588

Buffer size:    2144
Packet size:    2048
Trailer size:   12
Packet offset:  84
+
```

The fields that appear have the following meanings:

Interface Interface name.

- *Input buffers*

Req Number of input buffers required.

Alloc Number of assigned input buffers.

Low Low threshold for receive (input) buffers (flow control).

Curr Current number of input buffers on this device. If the value is 0, the device is disabled. When a packet is received, if the value of *Curr* is less than *Low*, then the packet is eligible for flow control. Consult the **queue** command to find out about the required conditions.

- *Buffer sizes*

Hdr Is the maximum value between the following two terms:

- largest LLC, plus MAC, plus the size of the device's output headers.

- MAC plus the size of the device's input headers.

Wrap Margin given for MAC, LLC or network layer.

Data Maximum data link layer packet size.

Trail Sum of the most extensive MAC and hardware trailers.

Total Overall size of each packet buffer. This value is calculated by adding the four fields indicated above.

- *Bytes*

Alloc Number of buffer memory bytes for this device. This value is calculated by multiplying the *Curr* value by the *Total* value.

Buffer size Full buffer size.

Packet size Maximum packet size.

Trailer size Maximum trailer size.

Packet offset Packet offset in the buffer.

3.2.3 CLEAR

Restarts the statistical information by clearing all counters on all interfaces. This is a useful command when you are looking for changes to large counters. However, it is important to note that this command does not save memory space or increase router speed.

Syntax:

```
+clear
```

- optional parameter **yes** allows the device to run an operation without prompting the user to confirm first. If this parameter is set to **yes**, no such confirmation is required. If not, the device prompts the user for confirmation.

Example:

```
+clear
Are you sure to clear stats?(Yes/No)?
+
```

Command history:

Release	Modification
11.01.06	The "[yes]" option was introduced as of version 11.01.06

3.2.4 CONFIGURATION

Displays information about the network protocols and interfaces. This command's output can be split into three sections. The first section contains the router ID, software version, boot ROM version, and watchdog status. It also contains the device's date and time settings and how much time has elapsed since the last restart. The second and third sections show information about the protocols that can be monitored and the interfaces present.

Syntax:

```
+configuration
```

Example:

```
+configuration

Teldat's Router, XXXXX 9 48  S/N: 524/00130
P.C.B.=91  Mask=0c10  Microcode=134f0  CLK=262144 KHz  BUSCLK=65536 KHz PCICLK=32768 KHz
ID: AT50-16F64R L9.48

DEMO licence active:
Licence will expire in 0 days 23 hours 53 minutes (base licence: 28 1013)

Boot ROM release:
 BIOS CODE VERSION: 01.10  Oct 30 2006 17:17:43
  gzip  Oct 30 2006 17:08:44
  io1  Oct 30 2006 17:17:36
  io2  Oct 30 2006 17:08:20
  io3  Oct 30 2006 17:17:36
 START FROM FLASH L1    Watchdog timer Enabled

Software release: 10.7.0 Nov 10 2006 15:20:04
Compiled by INTEGRATOR on INTEGRATOR2000
Loaded from primary partition

Hostname:                     Active user:
Date: Wednesday, 12/27/06     Time: 12:41:40
Router uptime: 9m4s
Num  Name     Protocol
0    IP       DOD-IP
3    ARP      Address Resolution Protocol
4    H323     H323
6    DHCP     Dynamic Host Configuration Protocol
11   SNMP     SNMP
13   RIP      Route Information Protocol
17   SIP      SIP
```

```
30   EAPOL    Extensible Authentication Protocol Over LAN
31   Preauth  WLAN Preauthentication

4 interfaces:
Connector     Interface           MAC/Data-Link        Status
GE0/FE0/LAN1  ethernet0/0         Ethernet/IEEE 802.3  Up
GE1/FE1/LAN2  ethernet0/1         Ethernet/IEEE 802.3  Testing
BRI/ISDN1     bri0/0              BRI Net              Testing
---           x25-node            internal             Up

SNMP OperStatus:
Interface       OperStatus
ethernet0/0     Up
ethernet0/1     Down
bri0/0          Down
x25-node        Up

Encryption Engines:
    Hardware: SEC-8272 Revision: 0xA, block 0x0

Cellular Driver Version: 00.09

WLAN Driver Version: 9.5.0.35.1
+
```

The first block (lines 1-3) displays general technical information about the device. The first line shows the type of router, the license in use and its serial number.

The second block (lines 4-5) displays information about the demo license, indicating the time remaining before the license expires, and the base license to which the device will return once the demo license expires or is manually disabled. This block only appears if there is an active demo license.

The third block (lines 6-12) shows the boot ROM (Read Only Memory) version that is currently installed on the router, the BIOS version, and the current watchdog timer configuration.

The fourth block (lines 13-15) shows the software version currently running on the router.

The fifth block (lines 16-18) shows the hostname, the active user, the date and time, and the time that has elapsed since the device was last restarted.

The sixth block shows a list of available protocols and interfaces. The meaning of each of the fields is as follows:

Num Number associated with the protocol.

Name Abbreviation for the protocol name.

Protocol Full name of the protocol.

The seventh block lists the interfaces available on the device. The meaning of each of the fields is as follows:

Connector Connector associated with the interface.

Interface Name of the interface.

MAC/Data Link Type of MAC/Data link configured for that interface.

Status Current status of the network interface.

Testing The interface is performing a self-diagnostic test. This happens when the router is first turned on and a problem is detected in the interface. Once the interface is on the network, it launches periodic test packets to ensure it is working properly. If a test fails, the router removes the interface from the network and runs a self-diagnostic test to ensure its integrity. If a fault occurs during a self-diagnostic test, the router declares the network out of service or down. If the self-diagnostic test completes successfully, the router declares the network up.

Up The interface is operational and connected.

Down The interface is not operational and a self-test failed. The router re-tests the network at increasing time intervals (starting with five seconds), until the router no longer tests the interface (which occurs after approximately two minutes).

Disabled The shutdown configuration command has disabled the interface.

Not present Either there is no interface present on the router or the console is incorrectly configured.

Unsupported The current version/license does not support the interface hardware.

Available The state of the secondary interface in a WAN configuration, when the primary interface is active.

Error Disabled. An interface error has been detected that caused it to be disabled.

The eighth block – **SNMP OperStatus** – shows the operating status of the interface from the point of view of the SNMP protocol, as defined in RFC 2233. The meaning of each of the fields is as follows:

Interface Name of the interface.

OperStatus The possible SNMP operating states are:

Up The interface is ready to pass and receive network traffic.

Down The interface is not operational.

Testing The interface is performing a self-diagnostic test so it cannot transmit real traffic packets.

Unknown For some reason, the interface's operating status can not be determined.

Dormant The interface is operational, but waiting for some external event to begin sending or receiving packets. Presumably it will switch to an ‘up’ state as soon as the expected event occurs. An example of this case is the dial-type interfaces when they have no traffic to carry (and therefore the call was not initiated) or when only incoming calls are allowed and the remote end has not started the connection.

Not present This state is a refinement of the ‘down’ state which indicates that the interface is down specifically because a component (typically hardware) is missing.

Lower layer down This is another refinement of the ‘down’ state, which in this case indicates that the interface extends from another interface(s) that is down.

The ninth block shows information about the encryption card and cellular/wireless LAN driver versions.

3.2.5 DEVICE

Displays statistical information about network interfaces, such as Ethernet, Token Ring, etc. This command can be used to obtain a summary of all interfaces or to obtain detailed information on a particular one.

Syntax:

```
+device [<interface>]
```

- **< interface >** is the name of the interface we want to display information about.

If no interface name is given, the system returns general information on all interfaces.

Example 1:

```
+device
                                    Auto-test   Auto-test     Maintenance
Interface            CSR    Vect      valids    failures        failures
ethernet0/0     fa200e00      27           1           0              0
serial0/0       fa200a00      5E           0         156              0
serial0/1       fa200a20      5D           0         156              0
serial0/2       fa200a60      5B           0           7              0
bri0/0          fa200a40      5C           1           0              0
x25-node               0       0           1           0              0
+
```

Example 2:

```
+device ethernet0/0
                                    Auto-test   Auto-test     Maintenance
Interface            CSR    Vect      valids    failures        failures
ethernet0/0     fa200e00      27           1           0              0

Physical address:  00A0267001E8
PROM address:      00A0267001E8
Speed:             10 Mbps

 Input statistics:
  failed, frame too long             0  failed, FCS error                  0
  failed, alignment error            0  failed, FIFO overrun               1
  internal MAC rcv error             1  packets missed                     1
 Output statistics:
  deferred transmission              0  single collision                   0
  multiple collisions                0  total collisions                   0
  failed, excess collisions          0  failed, FIFO underrun              0
  failed, carrier sense err          0  SQE test error                     0
  late collision                     0  internal MAC trans errors          0
  Ethernet MAC code release 1
+
```

The meaning of each field is as follows:

Interface Name of the interface.

CSR Command and Status Register.

Vect Vector regarding interruptions.

Auto Test Valids Number of times a link up is detected by the auto-test. It increases by one every time the link is detected. Whenever that happens, this check is interrupted until the link goes down.

Auto-Test Failures Number of times the auto-test does not detect the link established. When the link drops or is not established, the counter will increase by one for each self-test failed until the link establishes.

Maintenance Failures The maintenance test performs checks on the link at the physical level. The only parameter shown is the number of times a disconnection has ocurred. The counter will increase by one each time the link drops at the physical level, and will remain constant until the link establishes and drops again.

The aforementioned fields depend on the type of interface selected. Their names are self-explanatory when it comes to the information they provide.

Note

The screen shown may vary depending on the router and device.

3.2.6 ERROR

Displays network error statistics segmented for the various device interfaces. This command facilitates error counters.

Syntax:

```
+error
```

Example:

```
+error
                  Input      Input      Input      Input      Output      Output
Interface       Discards     Errors  Unk Proto  Flow Drop  Discards     Errors
ethernet0/0            0          0       1016          0           0          0
serial0/0              0          0          0          0           0          0
serial0/1              0          0          0          0           0          0
serial0/2              0          0          0          0           0          0
bri0/0                 0          0          0          0           0          0
x25-node               0          0          0          0           0          0
+
```

The meaning of each of the fields is as follows:

Interface Interface name.

Input Discards Number of packets discarded by flow control on reception.

Input Errors Number of packets that have been found to be faulty on the data link.

Input Unk Proto Number of packets received for an unknown protocol.

Input Flow Drop Number of received packets that have been subsequently discarded by transmission flow control.

Output Discards Number of packets discarded by transmission flow control.

Output Errors Number of output errors, such as attempts to send to a network that is down or that went down during transmission.

The sum between all the interfaces of *Input Flow Drops* and *Output Discards* is not equal because *Output Discards* can contain packets that are generated locally.

3.2.7 EVENT

Goes to the Event Logging System (ELS+) prompt and sets up temporary message filters for troubleshooting purposes. All changes made at the ELS+ prompt are immediate, but will disappear when the router is rebooted. For more information, see *Event Logging System ELS* on page 154. To return to the plus (+) prompt, type the**exit** command.

Syntax:

```
+event
```

Example:

```
+event
-- ELS Monitor --
ELS+
```

3.2.8 FEATURE

Provides access to the feature monitoring menu for features that are neither protocols nor network interfaces. Type a question mark (**?**) after the **feature** command to obtain a list of the features available for the software version.

This command is used to monitor the corresponding feature. For more information, check the relevant manual.

Note

Features must be enabled at the configuration prompt before they can be monitored.

Syntax:

```
+feature <option>
  access-lists             Generic IP lists monitoring
  afs                      Advanced firewall system feature
  bandwidth-reservation    Bandwidth-Reservation System feature monitoring
  dns                      DNS monitoring environment
  dns-updater              DNS UPDATER monitoring environment
  err-disable              Error disable monitoring
  gps-applications         GPS Applications monitoring
  hotspot                  Hotspot monitoring environment
  http                     HTTP server monitoring
  ip-discovery             Ip-discovery monitoring
  ipv6-access-list         IPv6 access list monitor
  istud                    IPSEC Tunnel Server Discovery Protocol monitoring
  ldap                     LDAP (Lightweight Directory Access Protocol)
                           monitoring
  mac-filtering            MAC-Filtering feature monitoring
  management               Management Monitoring
  management-platform      Management Platform monitoring
  netflow                  Netflow client monitoring
  nsla                     NSLA (Network Service Level Advisor) monitoring
  nsm                      NSM (Network Service Monitor) monitoring
  ntp                      NTP (Network Time Protocol) monitoring
  policy-map               Policy map monitoring
  power-switch             TeleControl Module control environment
  prefix-lists             Prefix lists monitoring
  radius                   RADIUS feature monitoring
  rmon                     RMON (Remote Network Monitoring)
  scada-forwarder          Scada protocol monitoring
  spi                      SPI agent monitoring
  stun                     Stun protocol monitoring
  syslog                   Syslog client monitoring
  tftp                     Access the device#s TFTP client.
  ttcp                     Ttcp (test tcp)
  vli                      Virtual Linux Interface monitoring
  wnms                     Wireless Network Management System monitoring
  wrr-backup-wan           WAN Reroute feature monitoring
+
```

- **< option >** specifies the type of information to display.

3.2.8.1 FEATURE ACCESS-LISTS

Accesses generic IP list monitoring.

Syntax:

```
+feature access-lists
```

Example:

```
+feature access-lists
-- Access Lists user console --
Access Lists+
```

For more information on how to monitor generic access lists, please see the following manual: *Teldat Dm752-I Access Control.*

3.2.8.2 FEATURE AFS

Accesses *AFS* monitoring.

Syntax:

```
+feature afs
```

Example:

```
+feature afs
-- AFS Monitor --
AFS+
```

For more information on *AFS* monitoring, please see the following manual: *Teldat Dm786-I AFS.*

3.2.8.3 FEATURE BANDWIDTH-RESERVATION

Accesses *Bandwidth-Reservation System* monitoring. For more information, please see the following manual: *Teldat Dm715-I Bandwidth Reservation System* .

System:

```
+feature bandwidth-reservation
```

Example:

```
+feature bandwidth-reservation
-- Bandwidth Reservation console --
BRS+
```

3.2.8.4 FEATURE DNS

Accesses the *DNS* monitoring environment. For more information, please see the following manual: *Teldat Dm723-I DNS Client*.

Syntax:

```
+feature dns
```

Example:

```
+feature dns
-- DNS resolver user console --
DNS+
```

3.2.8.5 FEATURE DNS-UPDATER

Accesses *dns-updater* monitoring.

Syntax:

```
+feature dns-updater
```

Example:

```
+feature dns-updater
-- DNS Updater console --
DNS Updater
```

For more information on *dns-updater* monitoring, please see the following manual: *Teldat Dm785-I DNS Updater.*

3.2.8.6 FEATURE ERR-DISABLE

Accesses *err-disable* monitoring.

Syntax:

```
+feature err-disable
```

Example:

```
+feature err-disable
-- Error Disable user console --
errdisable+
```

3.2.8.7 FEATURE FTP

Accesses *ftp* monitoring.

Syntax:

```
+feature ftp
```

Example:

```
+feature ftp

FTP Client

FTP+
```

For more information, please see the following manual: *Teldat Dm724-I FTP/sFTP Protocol.*

Command history:

Release	**Modification**
11.00.06	FTP option added.
11.01.02	FTP option added.

3.2.8.8 FEATURE GPS-APPLICATIONS

Accesses the *gps-applications* monitoring environment.

Syntax:

```
+feature gps-applications
```

Example:

```
+feature gps-applications
-- GPS Applications user console --
GPS Apps+
```

For more information on *gps-application* monitoring, please see the following manual: *Teldat Dm812-I GPS.*

3.2.8.9 FEATURE HOTSPOT

Accesses the *HotSpot* monitoring menu. For more information, please see the following manual: *Dm820-I HotSpot Feature*.

Syntax:

```
+feature hotspot
```

Example:

```
+feature hotspot
```

```
-- Hotspot User Console --
HS+
```

Command history:

Release	Modification
11.00.03	The "*Hotspot*" feature was introduced as of version 11.00.03.

3.2.8.10 FEATURE HTTP

Accesses *HTTP* protocol monitoring. For more information, please see the following manual: *Teldat Dm737-I HTTP Protocol.*

Syntax:

```
+feature http
```

Example:

```
+feature http
-- HTTP server user console --
HTTP+
```

3.2.8.11 FEATURE IP-DISCOVERY

Accesses the monitoring environment for the *ip-discovery* feature. This command (or functionality) is not available on all models.

Syntax:

```
+feature ip-discovery
```

Example:

```
+feature ip-discovery
-- TIDP Console --
TIDP+
```

3.2.8.12 FEATURE IPV6-ACCESS-LIST

Accesses *ipv6-access-list* monitoring.

Syntax:

```
+feature ipv6-access-list
```

Example:

```
+feature ipv6-access-list
-- IPv6 Access Lists user console --
IPv6 Access Lists+
```

For more information on *ipv6-access-list* monitoring, please see the following manual: *Teldat Dm808-I IPv6 Access Control.*

3.2.8.13 FEATURE ISTUD

Accesses *istud* monitoring.

Syntax:

```
+feature istud
```

Example:

```
+feature istud
-- ISTUD console --
ISTUD+
```

For more information about *istud* monitoring, please see the following manual: *Teldat Dm784-I ISTUD Feature.*

3.2.8.14 FEATURE LDAP

Accesses *LDAP* (Lightweight Directory Access Protocol) monitoring.

Syntax:

```
+feature ldap
```

Example:

```
+feature ldap
LDAP client monitor
LDAP+
```

3.2.8.15 FEATURE MAC-FILTERING

Accesses the *mac-filtering* monitoring environment.

Syntax:

```
+feature mac-filtering
```

Example:

```
+feature mac-filtering
-- MAC Filtering user console --
Filter+
```

3.2.8.16 FEATURE MANAGEMENT

Accesses the *management* monitoring environment.

Syntax:

```
+feature management
```

Example:

```
+feature management
--  Management Console  --
management+
```

Command history:

Release	Modification
11.00.05	The FEATURE MANAGEMENT menu was introduced as of version 11.00.05.
11.01.00	The FEATURE MANAGEMENT menu was introduced as of version 11.01.00.

3.2.8.16.1 SCRIPT <id_script>

Runs commands configured on the script.

Syntax:

```
management+script <id_script>
```

Example:

```
management+script 1
Starting executing Script 1
Sending command: p 4
Response: *
Response: Config>
Response: *
Command p 4 sent without errors
management+
```

Command history:

Release	Modification
11.00.05	The **script <id_script>** command was introduced as of version 11.00.05.
11.01.00	The **script <id_script>** command was introduced as of version 11.01.00.

3.2.8.16.2 STATISTICS

Lists all operations linked to advisor notifications, including information such as operation start times and whether any errors occurred during execution.

Syntax:

```
management+statistics
```

Example:

```
management+statistics

                 Management Operation Statistics
----------------------------------------------------------------------

----------------------------------------------------------------------

 ID      Count       Last Start Time           Errors        Name

----------------------------------------------------------------------

  2       1          2016-02-05 16:08:47           0        lte-up

  3       2          2016-02-05 16:09:00           0        lte-down

  4       2          2016-02-05 16:09:00           0        ip-config

  5       0          1970-01-01 00:00:00           0        prueba

  6       0          1970-01-01 00:00:00           0
```

Command history:

Release	Modification
11.00.05	The **statistics** command was introduced as of version 11.00.05.
11.01.00	The **statistics** command was introduced as of version 11.01.00.

3.2.8.17 FEATURE NETFLOW

Accesses *netflow* monitoring.

Syntax:

```
+feature netflow
```

Example:

```
+feature netflow
NETFLOW Monitor
NETFLOW Mon+
```

For more information, please see the following manual: *Teldat Dm789-I NETFLOW.*

3.2.8.18 FEATURE NSLA

Allows you to access the *NSLA* (Network Service Level Advisor) monitoring environment.

Syntax:

```
+feature nsla
```

Example:

```
+feature nsla
-- NSLA console --
NSLA+
```

For more information, please see the following manual: *Teldat Dm754-I NSLA (Network Service Level Advisor)* .

3.2.8.19 FEATURE NSM

Accesses the *NSM* (Network Service Monitor) monitoring environment.

Syntax:

```
+feature nsm
```

Example:

```
+feature nsm
-- NSM console --
NSM+
```

For more information, please see the following manual: *Teldat Dm749-I NSM* .

3.2.8.20 FEATURE NTP

Accesses the *NTP* (Network Time Protocol) monitoring environment. For more information, please see the following manual: *Teldat Dm728-I NTP Protocol* .

Syntax:

```
+feature ntp
```

Example:

```
+feature ntp
-- NTP user console --
NTP+
```

3.2.8.21 FEATURE POLICY-MAP

Accesses *policy-map* monitoring.

Syntax:

```
+feature policy-map
```

Example:

```
+feature policy-map
-- Policy Map user console --
Policy-Map+
```

For more information, please see the following manual: *Teldat Dm795-I Policy Map Class Map.*

3.2.8.22 FEATURE POWER-SWITCH

Accesses the control environment of the device's Telecontrol Module (MTC). This environment is only accessible to users via telnet and its use is only effective for devices connected to a power source through an MTC or an MTC+.

Syntax:

```
+feature power-switch
```

Example:

```
+feature power-switch
POWER-SWITCH monitor
POWER-SWITCH+
```

For more information, please see the following manuals: *Teldat Dm296-I Telecontrol Module (MTC)* ;*Teldat Dm689-I MTC+ Installation*.

3.2.8.23 FEATURE PREFIX-LISTS

Accesses *prefix list* monitoring.

Syntax:

```
+feature prefix-lists
```

Example:

```
+feature prefix-lists
-- Prefix Lists user console --
Prefix Lists+
```

For more information on *prefix list* monitoring, please see the following manual: *Teldat Dm780-I Prefix Lists.*

3.2.8.24 FEATURE RADIUS

Accesses *RADIUS* monitoring. For more information, please see the following manual: *Teldat Dm733-I RADIUS Protocol*.

Syntax:

```
+feature radius
```

Example:

```
+feature radius
-- RADIUS user console --
RADIUS+
```

3.2.8.25 FEATURE RMON

Accesses *rmon* monitoring.

Syntax:

```
+feature rmon
```

Example:

```
+feature rmon
-- RMON (Remote Network Monitoring) console --
RMON+
```

For more information on *rmon* monitoring, please see the following manual: *Teldat Dm796-I RMON Feature.*

3.2.8.26 FEATURE SCADA-FORWARDER

Accesses *SCADA forwarder* monitoring.

Syntax:

```
+feature scada-forwarder
```

Example:

```
+feature scada-forwarder
SCADA Forwarder Console
SCADA FWD+
```

3.2.8.27 FEATURE SPI

Accesses *spi* monitoring.

Syntax:

```
+feature spi
```

Example:

```
+feature spi
-- SPI Agent user console --
SPI+
```

For further information on how to configure *spi*, please see the following manual: *Teldat Dm793-I SPI Agent* .

3.2.8.28 FEATURE STUN

Allows you to access *STUN* client monitoring. For more information, please see the following manual: *Teldat Dm769-I STUN Protocol* .

Syntax:

```
+feature stun client
```

Example:

```
+feature stun client
STUN Client Monitor
STUN Client Mon+
```

3.2.8.29 FEATURE SYSLOG

Accesses *Syslog* client monitoring.

Syntax:

```
+feature syslog
```

Example:

```
+feature syslog
-- SYSLOG client console --
SYSLOG+
```

For more information, please see the following manual: *Teldat Dm753-I Syslog Client* .

3.2.8.30 FEATURE TFTP

Accesses *tftp* monitoring.

Syntax:

```
+feature tftp
```

Example:

```
+feature tftp
TFTP manager
TFTP+
```

For more information on *tftp* monitoring, please see the following manual: *Teldat Dm765-I TFTP Protocol.*

3.2.8.31 FEATURE TTCP

Provides access to a menu that performs *TCP* load testing.

Syntax:

```
+feature ttcp
```

Example:

```
+feature ttcp

-- TTCP --
Ttcp+
```

For more information on *TTCP* monitoring, please see the following manual: *Teldat Dm831-I TTCP Feature.*

3.2.8.32 FEATURE VLI

Accesses *vli* monitoring.

Syntax:

```
+feature vli
```

Example:

```
+feature vli
-- VLI monitor --
vli+
```

For more information on *vli* monitoring, please see the following manual: *Teldat Dm803-I Virtual Linux Interface (VLI).*

3.2.8.33 FEATURE WNMS

Allows you to access the *WNMS* (Wireless Network Management System) monitoring environment.

For more information, please see the following manual: *Teldat Dm819-I Wireless Network Management System* .

Syntax:

```
+feature wnms
```

Example:

```
+feature wnms
--  WNMS Console  --
WNMS+
```

Command history:

Release	Modification
11.00.03	The *WNMS* feature was instroduced as of version 11.00.03.

3.2.8.34 FEATURE WRR-BACKUP-WAN

Allows you to access *WRR-Backup WAN* monitoring. For more information, please see the following manual: *Teldat Dm727-I Backup WAN Reroute* .

Syntax:

```
+feature wrr-backup-wan
```

Example:

```
+feature wrr-backup-wan
-- Back-up WAN Reroute user console --
WRR+
```

3.2.9 HARDWARE

Allows you to modify the way hardware is configured. Enter a question mark (**?**) after the **hardware** command to obtain a list of available options.

Syntax:

```
+hardware <option>
  application     Set application hardware configuration
  show            Show hardware configuration
  switch          Set switch hardware configuration
+
```

Command history:

Release	Modification
11.00.07	The **hardware** command was introduced as of version 11.00.07.
11.01.02	The **hardware** command was introduced as of version 11.01.02.
11.01.04	The **application core** option was introduced as of version 11.01.04.

3.2.9.1 HARDWARE SHOW

Shows the current hardware configuration

Example:

```
+hardware show
  Current number of application cores is 2
  Switches are aggregated
```

Command history:

Release	Modification
11.01.04	The **application cores** option was introduced as of version 11.01.04.

3.2.9.2 HARDWARE SWITCH

Allows you to configure the hardware configuration of the switch. Entering a question mark (**?**) after the **hardware switch** command causes the device to list all the possible options.

```
+hardware switch ?
  aggregate    Interconnect switches in main board and SLOT1 to form a larger switch
  segregate    Segregate switch in SLOT1 from main board switch
+hardware switch
```

Command history:

Release	Modification
11.01.04	The **application cores** option was introduced as of version 11.01.04.

3.2.9.2.1 HARDWARE SWITCH AGGREGATE

Allows you to combine the expansion card switch with that of the main board.

When switches are not aggregated, the following text appears:

```
+hardware switch aggregate
  Switches are segregated
  --> Switches will be aggregated on next boot
```

Otherwise, the text is as follows and nothing happens:

```
+hardware switch aggregate
  Switches are aggregated
```

BIOS shows the next boot:

```
SWITCH(12) 10/100/1000
```

(Example with 8 Ethernet ports on the main board and a 4-port Ethernet card)

When switches are aggregated, they will appear as one switch with a single interface managing all ports. For more information, please see the relevant manual.

3.2.9.2.2 HARDWARE SWITCH SEGREGATE

Lets you separate the expansion card switch from that of the main board.

If switches are not segregated, the following text appears:

```
+hardware switch segregate
  Switches are aggregated
  --> Switches will be segregated on next boot
```

Otherwise, the text is as follows and nothing happens:

```
+hardware switch segregate
  Switches are segregated
```

On the next boot, switches will appear segregated and it is possible to configure switches as independent network interfaces with their relative number of ports. BIOS shows the following:

```
SWITCH(8) 10/100/1000 on main board
SWITCH(4) 10/100/1000 in socket 1
```

(Example with 8 Ethernet ports on the main board and a 4-Ethernet-port card)

3.2.9.3 HARDWARE APPLICATION

Allows you to set the applications-related hardware configuration. Entering a question mark (**?**) after the **hardware application** command causes the device to list all possible options.

```
+hardware application ?
  cores    Set number of application cores
+hardware application
```

Command history:

Release	Modification
11.01.04	The **application cores** option was introduced as of version 11.01.04.

3.2.9.3.1 HARDWARE APPLICATION CORES

Allows you to select the number of CPU cores used by the applications. The configurable core range allowed depends on the device hardware and the license that is currently loaded. Also, the configured number of cores affects the way the DRAM memory is allocated. For example, if the application cores option configured is 0, then all the available memory is assigned to the device firmware (CIT). However, if the application cores is > 0, then the total memory space is split between the space destined to the device firmware and the space destined to the applications.

Syntax:

```
+hardware application cores ?
  <0..2>    Value in the specified range
+
```

When we set a number of application cores different from the current one, the following text appears:

```
+hardware application cores 0
  Current number of application cores is 2
  --> Number of application cores will be 0 on next boot
+
```

Otherwise, the text is as follows and nothing happens:

```
+hardware application cores 2
  Current number of application cores is 2
+
```

When the number of application cores is 0, BIOS shows the next boot:

```
Mem Info:
DRAM size: 1024 Megabytes
   BANK 0: 1024 Megabytes
```

(There is only one bank, because all the memory is assigned to the device firmware)

When the number of application cores is > 0, BIOS shows the next boot:

```
Mem Info:
DRAM size: 1024 Megabytes
   BANK 0: 256 Megabytes
   BANK 1: 768 Megabytes
```

(Memory is split in two banks. One bank is assigned to the device firmware and the other one is assigned to the applications)

Command history:

Release	Modification
11.01.04	The **application cores** option was introduced as of version 11.01.04.

3.2.10 LAST-CONFIG-CHANGES

Allows you to monitor the latest changes made to the configuration. The first row shows the last configuration saved. Provided you haven't made any configuration changes since startup, this will be the active configuration. If some changes have been made, the active configuration appears in the second register. The first register is modified each time the configuration is saved without restarting the device.

Syntax:

```
+last-config-changes
```

Example:

```
+last-config-changes
Last configuration changes
-------------------------------------------------------------------------------
File     Acc-Type Address             User                 Date/Time
-------------------------------------------------------------------------------
```

```
SAMP1.CFG Console  0.0.0.0              CONSOLE             03/03/05  10:40:29
TFTP.CFG  Console  0.0.0.0              usersample           02/28/05  13:37:02
TFTP.CFG  Telnet   172.24.51.128        TELNET              02/24/05  15:29:40
TFTP.CFG  Telnet   172.24.51.128        TELNET              02/24/05  15:24:21
IGMP.CFG  Console  0.0.0.0              usersample           02/24/05  13:45:33
TFTP.CFG  Console  0.0.0.0              usersample           02/24/05  11:56:51
TFTP.CFG  Console  0.0.0.0              CONSOLE             02/24/05  11:54:51
IGMP.CFG  Console  0.0.0.0              usersample           02/23/05  19:34:36
IGMP.CFG  Console  0.0.0.0              CONSOLE             02/22/05  17:20:45
+
```

The meaning of the fields that appear is as follows:

File Name of the recorded and active file.

Acc-Type Type of access (telnet, console) used to modify the configuration.

Address IP address from which you accessed. This field will be 0.0.0.0 if the device was accessed by console.

User User who modified the configuration. If no users are defined on the device, this field will show the type of access.

Date/Time Date and time the changes were made.

3.2.11 LAST-APP-VERSION-CHANGES

Allows you to monitor the latest OS version upgrades. The first row shows the latest upgrade on the device. Only the last 15 upgrades are saved.

Syntax:

```
+last-app-version-changes
```

Example:

```
+last-app-version-changes

Last Application version changes
---------------------------------------------------
VERSION                 Date/Time
---------------------------------------------------
11.00.05-Beta-9ed5700   04/11/16  17:41:30
11.00.05-Beta-1e55c42   03/29/16  12:23:41
11.00.05-Beta-2d75a03   03/27/16  08:01:05
11.00.05-Beta-e853dde   03/20/16  21:15:42
11.00.05-Beta-9393dab   02/27/16  00:01:49
+
```

The meaning of each field is as follows:

VERSION Name of OS version.

Date/Time Date and time this version was loaded onto the device.

Command history:

Release	Modification
11.00.05	This command was introduced as of version 11.00.05.

3.2.12 LIST CURRENT-DEVICES

Lists information about available/configured interfaces.

Syntax:

```
+list current-devices
```

Example:

```
+list current-devices
Interface            Connector     Type of interface
```

```
ethernet0/0          GE0/FE0/LAN1  Fast Ethernet interface
ethernet0/1          GE1/FE1/LAN2  Fast Ethernet interface
bri0/0               BRI/ISDN1     ISDN Basic Rate Int
x25-node             ---           Router->Node
+
```

Command history:

Release	Modification
10.08.34.05.12	This command option was introduced as of version 10.08.34.05.12.
11.00.05	This command option was introduced as of version 11.00.05.
11.01.01	This command option was introduced as of version 11.01.01.

3.2.13 MALLOC-MONITOR

Allows you to monitor the device's internal memory management system. By using the command's options, you can control the system's memory management diagnostics. The use of this command requires a thorough knowledge of **Teldat Router** software and hardware architecture. Do not use this command unless instructed to do so by Teldat support personnel.

Two malloc-monitor tool versions are available: *verbose* and *builtin*. While the former records more information, the latter stores a log file if the device runs into an out-of-memory situation, making the information permanent. The *built-in* version is only available on certain devices, which have it enabled by default. The log file is called *memtrace.log* and can be displayed through the **malloc-monitor show-stored-log** command.

Syntax:

```
+malloc-monitor <option>
DISABLE
ENABLE
ASSIGNED-MEMORY-BLOCKS
LIST
SHOW-FLAGS
SHOW-STORED-LOG
ALL-REQUESTS
IGNORE-START-REQUESTS-FLAG
NEWEST-REQUESTS-FLAG (less than a minute age)
OLDEST-REQUESTS-FLAG (more than a minute age)
START-RECORD
STOP-RECORD
GET-MEMORY
VIEW-MEMORY
FREE-MEMORY
```

```
+malloc-monitor enable ?
  builtin    Built-in version of malloc monitor
  verbose    Verbose version of malloc monitor
  <cr>
+malloc-monitor enable ?
  builtin    Built-in version of malloc monitor
  verbose    Verbose version of malloc monitor
  <cr>
```

- **< option >** specifies the action to perform.

Command history:

Release	Modification
11.01.08	The **show-stored-log**, **enable builtin**, **enable verbose**, **disable builtin** and **disable verbose** command options were introduced as of version 11.01.08.

3.2.14 MANAGEMENT

Allows you to access the master router's monitoring environment.

Syntax:

```
                  +management
```

Example:

```
+management
-- Routers management user console--
MANAGEMENT+
```

3.2.15 MEMORY

Displays information on the different types of memory available on the device.

The command's output is organized in separate sections.

The first section, **system memory**, shows summary statistics and highlight the manner in which the system assigned RAM. These values are set when the router is booted and do not change while the device is in use.

The **system memory** section includes:

- *Total* : total amount of RAM installed in the router.
- *Dynamically managed*: amount of memory used for running processes.
- *Code*: fixed memory size, which is loaded with the system code.
- *Caches*: state of the router's cache memory devices.

In the **second section**, you can see information about **heap** memory. The data displayed in this section are divided into four extra sub-sections, separated by a blank line, providing a snapshot of **heap** memory status. The first sub-section is a summary of heap usage; the second sub-section lists the current chunks (a size smaller than block size); the third sub-section provides statistics on chunks that are one size larger than block size. Their statistics are stored in registers, labeled in arithmetic progression, based on their chunk size; the fourth sub-section provides statistics on chunks that are a size bigger than those displayed in the previous sub-section. Statistics on the latter are stored in registers, labeled in geometric progression, based on their chunk size.

The first sub-section of the **heap** shows the following fields:

- *Total* : amount of memory assigned to the **heap**, including internal system data.
- *Usable*: maximum amount of memory the **heap** can allocate to support running processes.
- *Free* : free memory in the **heap**.
- *Block size*: the **heap** is divided into **n** sized blocks.

The second sub-section of the **heap** shows the following data:

```
64 bytes chunks:    7104 (    454656 bytes) total,      10 (       640 bytes) free
   Chunk usage:  99.00%
```

The meaning of each column is as follows:

- Size of registered chunk.
- Total number of assigned chunks of this size in the **heap**.
- Total number of assigned bytes for this chunk size in the **heap**.
- Total number of free chunks of this size in the **heap**.
- Total number of bytes of free chunks of this size in the **heap**.

Chunk usage is a global measurement used for chunks that are currently assigned and are smaller than the block size.

In this second sub-section, a number of assigned chunks, for a given size, is allocated as needed by taking memory space from the chunks displayed in subsequent sub-sections. The more chunks in this sub-section, the less size for chunks in subsequent sub-sections.

The third and fourth sub-sections of the **heap** show the following data:

```
12288 bytes chunks:     85 (   1044480 bytes) used (    85 max),      0 (          0 bytes) free

16822272 bytes chunks:      2 (  54730752 bytes) used (     2 max),      0 (          0 bytes) free
```

The meaning of each column is as follows:

- Size of registered chunk.
- Total number of assigned chunks of this size in the **heap**.

- Total number of assigned bytes for this chunk size in the **heap**.
- Maximum number of assigned chunks of this size concurrently allocated in the **heap**.
- Total number of free chunks of this size in the **heap**.
- Total number of bytes of free chunks of this size in the **heap**.

As already mentioned, it is important to understand that statistics for chunk sizes in the third sub-section are registered in arithmetical progression, while in the fourth sub-section they are stored geometrically. This means that, in the third sub-section, the number of assigned bytes for a chunk size is calculated by multiplying the chunk size by the number of chunks, and this does not match the fourth sub-section.

Chunk sizes for registers in the third sub-section are separated by block size. The first line in the example is interpreted as follows: the register with the 12288 byte-sized chunks, has 85 allocated chunks with a total size of 1044480 bytes. The maximum number of concurrently assigned chunks for this register is 85 and there are no free chunks of 12288 bytes to be stored in this register.

Chunk sizes for registers in the fourth sub-section are separated by a multiple of block size, resulting in the evolution of registered chunk sizes in a geometrical progression form. Here, each register shares statistics between different chunk sizes; this is what happens in the second line of the previous example. This line is interpreted as follows: in the register for chunk sizes between 16822272 bytes and the next registered chunk size, there are two allocated chunks with a total size of 54730752 bytes. The maximum number of chunks concurrently assigned for this register is two and there are no free chunks with said size to be stored in this register.

Please note that statistics relating to different chunk sizes may not be available on your device model. If this is the case, global information on used and free chunks is displayed instead.

The **third section** displays information relating to memory **pool1**. **pool1** is an area of memory broken up into fixed partitions reserved for messages and buffers from the node.

The following information is displayed for **pool1**:

- *Sz*: total size of the POOL in bytes.
- *AllocPart*: number of partitions in use.
- *AvlPart*: number of available partitions.

The **fourth section** shows parameters relating to:

- *Flash memory*: system flash memory measured in bytes.
- *Free global Buffers*: number of public buffers available in the system. The minimum number of available public buffers reached is shown in brackets.
- *Orphan data buffers*: number of data buffers available to be assigned to a global buffer. They are called *orphan* because they have not yet been assigned.
- *Spurious INT 47*: spurious interruption counter in the system.

Syntax:

```
+memory ?
  history-48h    Show 48h mem history
  <cr>
+memory history-48h [<max_hour>] [<min_hour>]
```

Example:

```
+memory

SYSTEM MEMORY:
   Total                  536870912
   Dynamically managed    502296576
   Code                    34574336
   Caches  ON    Write Back

HEAP:
   total                  502296576
   usable                 501772288
   free                   377067688
   block size                  4096

            8 bytes chunks:   39936 (    319488 bytes) total,    345 (      2760 bytes) free
           16 bytes chunks:    5888 (     94208 bytes) total,    244 (      3904 bytes) free
           32 bytes chunks:   35072 (   1122304 bytes) total,    105 (      3360 bytes) free
           64 bytes chunks:    7168 (    458752 bytes) total,     24 (      1536 bytes) free
```

```
          128 bytes chunks:     544 (     69632 bytes) total,     11 (      1408 bytes) free
          256 bytes chunks:      80 (     20480 bytes) total,      4 (      1024 bytes) free
          512 bytes chunks:   28968 (  14831616 bytes) total,      1 (       512 bytes) free
         1024 bytes chunks:      56 (     57344 bytes) total,      0 (         0 bytes) free
         2048 bytes chunks:   36934 (  75640832 bytes) total,      0 (         0 bytes) free
   Chunk usage:  99.00%

         4096 bytes chunks:  22638 (  92725248 bytes) used ( 22638 max),      0 (         0 bytes) free
         8192 bytes chunks:     28 (    229376 bytes) used (    28 max),      0 (         0 bytes) free
        12288 bytes chunks:     85 (   1044480 bytes) used (    85 max),      0 (         0 bytes) free
        16384 bytes chunks:     45 (    737280 bytes) used (    45 max),      0 (         0 bytes) free
        20480 bytes chunks:     10 (    204800 bytes) used (    10 max),      0 (         0 bytes) free
        24576 bytes chunks:      1 (     24576 bytes) used (     1 max),      0 (         0 bytes) free
        32768 bytes chunks:      1 (     32768 bytes) used (     1 max),      0 (         0 bytes) free
        36864 bytes chunks:     13 (    479232 bytes) used (    13 max),      0 (         0 bytes) free

        61440 bytes chunks:      5 (    331776 bytes) used (     5 max),      0 (         0 bytes) free
       110592 bytes chunks:      1 (    151552 bytes) used (     1 max),      0 (         0 bytes) free
       176128 bytes chunks:      2 (    503808 bytes) used (     2 max),      0 (         0 bytes) free
       307200 bytes chunks:      1 (    524288 bytes) used (     1 max),      0 (         0 bytes) free
     16822272 bytes chunks:      1 (  27729920 bytes) used (     1 max),      0 (         0 bytes) free
     67153920 bytes chunks:      0 (         0 bytes) used (     1 max),      0 (         0 bytes) free
    268480512 bytes chunks:      0 (         0 bytes) used (     0 max),      1 ( 377053184 bytes) free

POOLS:
   POOL1    Sz:9631710    AllocPart:1          AvlPart:27917

Flash Memory: 67108864

Free global Buffers:36261 (36258)

Orphan Data Buffers(maximum reached):36261 (36873)

Spurious INT 47: 0
+
```

Run the **memory history-48h** command to display a graph showing the status of free memory from the last 48 hours:

```
+memory history-48h
Free memory percentage during the last 48 hours (%):

     666666666666666666666666666666666666666666666666666666666666
100_ 999999999999999999999999999999999999999999999999999999999999
 95_|
 90_|
 85_|
 80_|
 75_|
 70_|
 65_|............................................................
 60_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 55_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 50_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 45_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 40_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 35_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 30_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 25_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 20_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 15_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
 10_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
  5_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
  0_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
   48h.           36h            24h            12h             0h
```

In this example we can see that memory usage was stable during the last 48 hours, with 69% of the available free memory used. The available command options enable you to select an hour range for which to graph data.

> **Important**
>
> The command's output may vary slightly according to your router model.

Command history:

Release	Modification
11.01.08	The **memory history-48h** command option was introduced as of version 11.01.08.

3.2.16 NETWORK

Displays the monitoring prompt for the specified network interface (supported networks include Frame Relay, PPP, X.25, etc.). From the prompt, you can display statistical information.

Syntax:

```
+network <interface>
```

- **< interface >** is the name of the network interface whose monitoring environment we want to access.

Type the **device** command at the plus prompt (**+**) for a list of the networks for which the router is configured.

Example:

```
+device
                                    Auto-test   Auto-test     Maintenance
Interface            CSR    Vect      valids    failures        failures
ethernet0/0     fa200e00      27           1           0               0
serial0/0       fa200a00      5E           0         373               0
serial0/1       fa200a20      5D           0         373               0
serial0/2       fa200a60      5B           0          10               0
bri0/0          fa200a40      5C           1           0               0
x25-node               0       0           1           0               0
+network serial0/0

-- Frame Relay Console --
serial0/0 FR+
```

For more information, check the relevant network interface manual.

3.2.17 NODE commands

Allows you to access the node monitoring environment (X.25, XOT and 270).

Syntax:

```
+node <name>
X25
XOT
270
```

- **< name >** is the name of the node whose monitoring menu we want to access**.**

3.2.17.1 NODE X25

Allows you to access the X.25 node monitoring environment. Port parameters can be configured in the network port (+**networkport** command).

Syntax:

```
+node x25
```

Example:

```
+node x25
-- X25 Monitor --
X25+
```

For more information on the X.25 node configuration environment, please see the following manual: *Teldat Dm707-I*

X.25.

3.2.17.2 NODE 270

Allows you to access the 270 node monitoring environment. You can also gain access from the 270 network (+ **network port** command).

Syntax:

```
+node 270
```

Example:

```
+node 270
270 Monitoring
270>
```

3.2.18 PROTOCOL

Allows you to access the command environment of the software protocols installed on your router. Typing the **protocol** command followed by the desired protocol number, or short name, accesses a protocol's command environment. After you enter this command, the prompt of the specified protocol appears. From the prompt, you can enter commands that are specific to that protocol.

To enter a protocol's command environment:

(1) Type the **protocol ?** command to see the list of protocols configured on the router.

(2) Enter the desired protocol name. The specified protocol prompt will appear immediately. From this prompt, you can start typing commands specific to that protocol.

(3) To return to the **+** prompt, type **exit**.

Syntax:

```
+protocol <identifier>
```

- **< identifier >** is the name of the protocol whose monitoring menu we want to access.

Example:

```
+protocol ?
  arp      Access ARP protocol
  asrt     Access ASRT protocol
  bfd      Access BFD protocol
  bgp      Access BGP protocol
  dhcp     Access DHCP protocol
  dhcpv6    Access DHCPv6 protocol
  dls      Access DLS protocol
  dot1x    Access 802.1X protocol
  gw104    Access GW-104 protocol
  h323     Access H323 protocol
  igmp     Access IGMP protocol
  ip       Access IP protocol
  ipv6     Access IPv6 protocol
  l2tp     Access L2TP protocol
  mgcp     Access MGCP protocol
  nhrp     Access NHRP protocol
  noe      Access NOE protocol
  ospf     Access OSPF protocol
  pim      Access PIM protocol
  rip      Access RIP protocol
  ripng    Access RIPNG protocol
  sccp     Access SCCP protocol
  sip      Access SIP protocol
  snmp     Access SNMP protocol
+protocol ip
-- IP protocol monitor --
IP+
```

3.2.19 QUEUE

Shows statistics about the input and output queues on the specified interfaces. Information provided by the **queue** command includes:

- The total number of buffers allocated.
- The maximum number of buffers processed in the burst.
- The low-level buffer value.
- The number of buffers currently active on the interface.
- The minimum or maximum number of buffers in the interface.

Syntax:

```
+queue <interface>
```

- **< interface >** is the name of the network interface whose monitoring environment we want to access.

If no interface is specified, information about all of the available interfaces is displayed.

Example:

```
+queue
                                     Input Queue                             Output Queue
Interface          Alloc  Burst   Low  Curr   Min   Cur.Req  Max.Req      Fair   Current   Max
ethernet0/0          256     16     5   256   256      256      257       246        0        0
ethernet0/1          256     16     5   256     0      256      256       246        0        0
x25-node               0    ---     0   100     0        0        0        40        0        0
ethernet1/0           40     16     5    40    40       40       41        40        0        0
+
```

The meaning of the fields that appear is as follows:

Interface Interface name.

- *Input Queue*

Alloc Number of buffers assigned to this device.

Burst Maximum number of buffers processed in the burst. In high-load situations, several buffers are processed in the burst to increase performance, thus achieving higher speeds. This information is only available on some interfaces.

Low Low water mark for flow control on this device.

Curr Current number device buffers. The value will be 0 if the device is disabled.

Min Minimum number of buffers in this device's input queue. As of version 11.01.01, this field is no longer available.

Cur.Req Number of buffers currently used by the interface for reception. It is the sum of the buffers ready for reception and the buffers used for reception that are currently being processed by the router.

Max.Req Maximum number of buffers in this device's output queue.

- *Output Queue*

Fair Fair level for the length of the output queue on this device.

Current Number of packets currently waiting to be transmitted on this device.

Max Maximum number of buffers in the device's output queue. As of version 11.01.01, this field is no longer available.

The router attempts to keep at least the *Low* value packets available for receiving over an interface. If a packet is received and the value of *Curr* is less than *Low*, then the packet will be subject to flow control. If a buffer subject to flow control is to be queued on this device and the *Curr* level is greater than *Fair*, the router drops the buffer instead of placing it in the queue. The dropped buffer is displayed in the Output Discards column of the **error** command. It will also generate ELS event GW.036 or GW.057.

Command history:

Release	Modification
11.01.01	The **Min** and **Max** fields are obsolete as of version 11.01.01.

3.2.20 QUICK

To access the quick monitoring menu you must first access the general monitoring menu and type quick. This command accesses the quick monitoring menu.

Syntax:

```
+quick
```

Example:

```
+quick
Quick Monitor Menu
Quick Monitor+
```

Command history:

Release	Modification
11.00.02	This command is obsolete as of version 11.00.02. The quick menu configuration is no longer supported.

3.2.21 RUSH-ENGINE

Displays statistical information about the rush engine and the existing flows on each interface.

Syntax:

```
+rush-engine ?
  all          Monitor all interfaces
  interface    Monitor interface information
  state        Display Rush Engine state
+
```

Command history:

Release	Modification
11.01.00	This command was introduced as of version 11.01.00.

3.2.21.1 RUSH-ENGINE ALL

Displays global state and statistical information on all interfaces. You can set up a filter to display only the flows that match that filter.

This command may also display only a summary of the statistics, clear current counters and remove all flows from the rush engine.

Syntax:

```
+rush-engine all [summary | <filter> | clear | flush]
<filter>:
      ipv4 [destination-address <dstaddr>]
           [source-address <srcaddr>]
           [address <addr>]
      ipv6 [destination-address <dstaddr>]
           [source-address <srcaddr>]
           [address <addr>]
      protocol {tcp [destination-port <dstport>] [source-port <srcport>] [port <port>] |
                udp [destination-port <dstport>] [source-port <srcport>] [port <port>] |
    gre}
+
```

ipv4		Only selects IPv4 protocol flows.
	destination-address	Only selects flows whose destination address matches the one referenced.
	source-address	Only selects flows whose source address matches the one referenced.
	address	Only selects flows whose source or destination address matches the one referenced.

ipv6		Only selects IPv6 protocol flows.
	destination-address	Only selects flows whose IPv6 destination address matches the one referenced.
	source-address	Only selects flows whose IPv6 source address matches the one referenced.
	address	Only selects flows whose IPv6 source or destination address matches the one referenced.
protocol		Only selects flows for the referenced protocol.
	tcp	Only selects flows for the TCP protocol.
	udp	Only selects flows for the UDP protocol.
	destination-port	Destination port for TCP or UDP flows.
	source-port	Source port for TCP or UDP flows.
	port	Source or destination port for TCP or UDP flows.
	gre	Only selects flows for the GRE protocol.

Example:

Lists TCP flows in all interfaces.

```
+rush-engine all protocol tcp

Rush Engine is Enabled

Last cache invalidation 1m38s ago
Idle flow timeout 5
Flow stats burst 32
Memory sizes: RBD 20, Ifc 15944, Flow 204, Flow add 144, Max flows 2644

Rush Ifc ethernet0/0:
        active flows    5/9     hash collisions 0/0
        in  packets        172  in  bytes      151076
        out packets        153  out bytes      110533
        stats cleared 0 times
   Active flows details (27fb000):
        key 0da0 age 0 #1 TCP 64.233.184.188:5228 -> 192.168.212.245:52791 tos 0/0 mtu 1500 NAT/
 (->e9e2000:out) stats 7/924
        key 0ca4 age 6000 #1 TCP 173.192.222.189:443 -> 192.168.212.245:36265 tos 0/0 mtu 1500 NAT/
 (->e9e2000:out) stats 6/663
        key 03cf age 10000 #1 TCP 216.58.211.238:443 -> 192.168.212.245:46999 tos 0/0 mtu 1500 NAT/
 (->e9e2000:out) stats 15/6028
        key 0631 age 8000 #1 TCP 216.58.211.238:443 -> 192.168.212.245:60657 tos 0/0 mtu 1500 NAT/
 (->e9e2000:out) stats 50/44250
        key 0870 age 0 #2 IPv6 TCP 1::2[54174] -> 2::2[5001] tos 0/0 label 729962 mtu 1500/
 (->e9e2000:out) stats 283217/424824096

Rush Ifc wlan0/0:
        active flows    5/6     hash collisions 0/0
        in  packets        153  in  bytes      110533
        out packets        172  out bytes      151076
        stats cleared 0 times
   Active flows details (e9e2000):
        key 0dac age 0 TCP 192.168.1.31:52791 -> 64.233.184.188:5228 tos 0/0 mtu 1500 NAT/
 (->27fb000:out) stats 4/784
        key 0207 age 6000 TCP 192.168.1.31:36265 -> 173.192.222.189:443 tos 0/0 mtu 1500 NAT/
 (->27fb000:out) stats 7/733
        key 0e03 age 10000 TCP 192.168.1.31:46999 -> 216.58.211.238:443 tos 0/0 mtu 1500 NAT/
 (->27fb000:out) stats 11/3351
        key 0561 age 8000 TCP 192.168.1.31:60657 -> 216.58.211.238:443 tos 0/0 mtu 1500 NAT/
 (->27fb000:out) stats 34/6397
        key 0f94 age 0 IPv6 TCP 2::2[5001] -> 1::2[54174] tos 0/0 mtu 1500 (->27fb000:out) stats 22974/1658328
+
```

Command history:

Release	Modification
11.01.00	This command option was introduced as of version 11.01.00.
11.01.04	A command option to filter via the IPv6 protocol was included. Information on the input port in switches and input/output sub-interfaces is displayed.
11.01.06, 11.01.05.30.01	New items have been added to the monitoring command's output: the time elapsed since the last cache invalidation and flow statistics update bursts.
11.01.09	The "*summary*" command option was introduced as of version 11.01.09.

3.2.21.2 RUSH-ENGINE INTERFACE

Displays information on one interface. You can specify a filter to display only the flows that match that filter.

This command may also display a summary of the statistics, clear current counters and remove all flows in the interface from the rush engine.

Syntax:

```
+rush-engine interface <interface> [summary | <filter> | clear | flush]
<filter>:
      ipv4 [destination-address <dstaddr>]
           [source-address <srcaddr>]
           [address <addr>]
      ipv6 [destination-address <dstaddr>]
           [source-address <srcaddr>]
           [address <addr>]
      protocol {tcp [destination-port <dstport>] [source-port <srcport>] [port <port>] |
                udp [destination-port <dstport>] [source-port <srcport>] [port <port>] |
    gre}
+
```

ipv4		Only selects IPv4 protocol flows.
	destination-address	Only selects flows whose destination address matches the one referenced.
	source-address	Only selects flows whose source address matches the one referenced.
	address	Only selects flows whose source or destination address matches the one referenced.
ipv6		Only selects IPv6 protocol flows.
	destination-address	Only selects flows whose IPv6 destination address matches the one referenced.
	source-address	Only selects flows whose IPv6 source address matches the one referenced.
	address	Only selects flows whose IPv6 source or destination address matches the one referenced.
protocol		Only selects flows for the referenced protocol.
	tcp	Only selects flows for the TCP protocol.
	udp	Only selects flows for the UDP protocol.
	destination-port	Destination port for TCP or UDP flows.
	source-port	Source port for TCP or UDP flows.
	port	Source or destination port for TCP or UDP flows.
	gre	Only selects flows for the GRE protocol.

Example:

Remove all flows from the *ethernet0/0* interface.

```
+rush-engine interface ethernet0/0 flush
+
```

Command history:

Release	Modification
11.01.00	This command option was introduced as of version 11.01.00.

Release	Modification
11.01.04	A command option to filter via the IPv6 protocol was included. Information on the input port in switches and input/output sub-interfaces is displayed as of version 11.01.04.
11.01.09	The "*summary*" command option was introduced as of version 11.01.09.

3.2.21.3 RUSH-ENGINE STATE

Displays the rush engine status.

Shows the rush engine status, the time elapsed since the last cache invalidation, the current idle flow timeout value, the burst length for flow statistics updates, and information relating to the amount of memory used by internal structures.

Syntax:

```
+rush-engine state

Rush Engine is Enabled

Last cache invalidation 3m24s ago
Idle flow timeout 5
Flow stats burst 32
Memory sizes: RBD 20, Ifc 15944, Flow 204, Flow add 144, Max flows 2644
+
```

Command history:

Release	Modification
11.01.00	This command option was introduced as of version 11.01.00.
11.01.06, 11.01.05.30.01	New items have been added to the monitoring command's output: the time elapsed since the last cache invalidation and flow statistics update bursts.

3.2.22 STATISTICS

Displays statistical information about the network software, such as the configuration of the networks in the router.

Syntax:

```
+statistics <interface>
```

- **< interface >** is the name of the network interface whose monitoring environment we want to access.

To see the networks for which the router is configured, type the **device** command at the **+** prompt. If no network interface is specified, information about all of the available networks on the device is displayed.

Example:

```
+statistics
                    Unicast  Multicast      Bytes       Packets         Bytes
Interface          Pqts Rcv   Pqts Rcv   Received   Transmitted   Transmitted
ethernet0/0               0       5384    3090255             0             0
serial0/0                 0          0          0             0             0
serial0/1                 0          0          0             0             0
serial0/2                 0          0          0             0             0
bri0/0                    0          0          0             0             0
x25-node                  0          0          0             0             0
+
```

The meaning of the fields that appear is as follows:

Interface Interface name.

Unicast Pkts Rcv Number of non-multicast, non-broadcast specifically-addressed packets at the MAC layer.

Multicast Pkts Rcv Number of multicast or broadcast packets received.

Bytes Received Number of bytes received at the MAC layer.

Packets Trans Number of unicast, multicast, or broadcast packets transmitted.

Bytes Trans Number of bytes transmitted at the MAC layer.

3.2.23 SYSTEM

Allows you to monitor system memory, stacks and CPU usage; configure the console port speed; display the firmware needed for the proper functioning of the device; activate certain debugging information; display user login history; view open Telnet and SSH sessions; and exchange commands or messages between the terminals corresponding to those open sessions.

Syntax:

```
+system <option>
  ap-embedded                Embedded AP control actions
  console-speed              Configure the console-serial port speed.
  cpu-graph                  Display a system load measurement graph
  cpu-history-48h            Display the system load for the last 48 hours
  cpu-text                   Display the average load in the system
  dba-debug                  DBA subsystem debug level
  disable-process-monitor    Disable CPU load monitoring of processes
  enable-process-monitor     Enable CPU load monitoring of processes
  firmwares-required         Display the firmware required
  health                     Display information about equipment status
  http                       Display information on the users connected by HTTP
  licence                    Show information about licences in the equipment
  login                      Shows if the difference between upper and lower
                             case characters is activated
  login-historic             Display a list containing information on the users
                             who have accessed the management console
  memory                     Display statistics on the system memory
  pcmcia                     Access the PCMCIA interface status monitoring
  power-off-status           Display power off timers and ignition status
  process-list               Display the system processes status
  ssh                        Display information on the users connected by SSH
  stack-status               Display the system stack status
  telnet                     Display information on the users connected to the
                             device
  telnet-clients             Display information on open Telnet sessions to
                             remote devices
  terminal                   Interchange commands or messages between terminals
  usb                        Access the USB interface status monitoring
+
```

- **< option >** specifies the action to take.

3.2.23.1 SYSTEM AP-EMBEDDED

Performs actions over the embedded Wi-Fi access-point (only possible in devices where this is available).

Syntax:

```
+system ap-embedded ?
  factory-reset    Perform a factory reset
  hard-reset       Perform a hard reset
+system ap-embedded
```

Command history:

Release	Modification
11.01.08	The "*system ap-embedded*" command option was introduced as of version 11.01.08.

3.2.23.1.1 SYSTEM AP-EMBEDDED FACTORY-RESET

Allows a user to perform a factory reset (set the factory-default configuration).

The user must make sure the AP is not writing in its Flash memory. Otherwise, it can become corrupted.

Example:

```

+system ap-embedded factory-reset
About to factory reset the embedded AP.
```

```
This can cause serious damage to the AP.
Are you sure(Yes/No)? Yes
This takes a while: be patient.
Proceeding...Done!
+
```

Command history:

Release	Modification
11.01.08	The "*system ap-embedded factory-reset*" suboption was introduced as of version 11.01.08.

3.2.23.1.2 SYSTEM AP-EMBEDDED HARD-RESET

Allows a user to perform a hard reset (power off/power on cycle).

The user must make sure the AP is not writing in its Flash memory. Otherwise, it can become corrupted.

Example:

```
+system ap-embedded hard-reset
About to hard reset the embedded AP.
This can cause serious damage to the AP.
Are you sure(Yes/No)? Yes
Proceeding...Done!
+
```

Command history:

Release	Modification
11.01.08	The "*system ap-embedded hard-reset*" suboption was introduced as of version 11.01.08.

3.2.23.2 SYSTEM CONSOLE-SPEED

Configures the console serial port speed.

Syntax:

```
+system console <speed>
```

< speed > is the speed in bits per second (bps) to set. Valid values are 9.600, 14.400, 19.200, 38.400, 57.600 and 115.200.

Example:

```
+system console 9600
+
```

The console serial port speed is set at **9600 bps** by default.

3.2.23.3 SYSTEM CPU-GRAPH

Displays a graph of the percent CPU usage over the last five minutes. The data runs from left to right with the oldest values to the left and the most recent to the right. Each column represents the load over 5 seconds, and is displayed both in graphical (a vertical bar of points that ascends to the corresponding percentage) and in numerical format (written vertically from top to bottom).

Syntax:

```
+system cpu-graph
```

Example:

```
+system cpu-graph
CPU usage during the last 5 minutes (%):

     211122222211111111111111111111111111111111111111111111111
100_ 588837877797797777777777777777777777777777777777777777777
```

```
 90_|
 80_|
 70_|
 60_|
 50_|
 40_|
 30_|
 20_|.   ..:...
 10_|:::::::::::..:..............................................
  0_|::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
  5min.                                           1min.         0

+
```

In this example, we can see that five minutes ago (left column) there was a 25 % load (vertical reading at the top of that column). This load has been changing over time (18 % in the next 15 seconds, 23 % in the following 5 seconds, etc.) until stabilizing at the current 17 % load (right column).

3.2.23.4 SYSTEM CPU-HISTORY-48H

Provides a graph of the system's CPU usage over time (up to a maximum of 48 hours). The data runs from left to right with the oldest values in time to the left and the most recent to the right. The CPU load values read vary depending on the time range displayed on the graph's horizontal axis. Thus, a 2-hour time range shows the maximum CPU usage percentages of each 2-minute interval, and a 48-hour time range shows the maximum percentages of each 48-minute interval. The numerical percentage values are at the top of the graph and are read vertically from top to bottom. The command allows you to specify a time range in viewing hours within the available history.

Syntax:

```
+system cpu-history-48h [<max_hour>] [<min_hour>]
```

The parameters to configure are:

max_hour Left limit, in hours, of the horizontal axis range.

min_hour Right limit, in hours, of the horizontal axis range.

Example 1:

```
+system cpu-history-48h
CPU usage during the last 48 hours (%):
                                      21
100_ 225222222222222222222222222222229262222222222222222223242222 2
 95_|
 90_|
 85_|
 80_|
 75_|
 70_|
 65_|
 60_|
 55_|
 50_|
 45_|
 40_|
 35_|
 30_|
 25_|
 20_|                                 .
 15_|                                 :
 10_|                                 ::
  5_|                                 :::
  0_|..:.............................:::........................
   48h.           36h            24h            12h            0h

+
```

In this example, all the available history up to the current time is displayed because no input parameters have been specified. The time range shown is 3 to 0 hours because the device has not been in operation for more than three hours. In this way, each percentage value in the graph represents the maximum CPU usage over a 3-minute interval. The range of the horizontal axis will automatically increase while the device stays in operation, up to a maximum of

48 hours. After this time, the axis will remain in the range between 48 and 0 hours.

Example 2:

```
+system cpu-history-48h 12 3
CPU usage in range from 12 to 3 hours ago (%):
100_ 222222222322222222222222222222592222222222222222222223124222
 95_|
 90_|
 85_|
 80_|
 75_|
 70_|
 65_|
 60_|
 55_|
 50_|
 45_|
 40_|
 35_|
 30_|
 25_|
 20_|
 15_|
 10_|
  5_|                                  :                     :
  0_|.........:.......................:.....................:.....
   12h.                11h                  7h                  3

+
```

In this example, the *max_hour* and *min_hour* parameters have been specified. They determine that the display time ranges from 12 to 3 hours prior.

3.2.23.5 SYSTEM CPU-TEXT

Shows the short, medium, and long-term CPU usage as a percentage.

Syntax:

```
+system cpu-text
```

Example:

```
+system cpu-text
CPU Short-Term Usage (5 sec.):   24.5%
CPU Medium-Term Usage (1 min.):  19.2%
CPU Long-Term Usage (5 min.):    17.6%
+
```

3.2.23.6 SYSTEM DISABLE-PROCESS-MONITOR

Disables system process monitoring, thus preventing you from obtaining CPU usage statistics by process. Process monitoring is disabled when the device is booted.

Syntax:

```
+system disable-process-monitor
```

Example:

```
+system disable-process-monitor
Process monitoring disabled
+
```

3.2.23.7 SYSTEM ENABLE-PROCESS-MONITOR

Enables system process monitoring, allowing you to obtain CPU usage statistics by process. Process monitoring is disabled when the device is booted.

Syntax:

```
+system enable-process-monitor
```

Example:

```
+system enable-process-monitor
Process monitoring enabled
Equipment performances can be affected while process monitoring is enabled.
+
```

Note

The device's performance may be affected when process monitoring is enabled.

3.2.23.8 SYSTEM FIRMWARES-REQUIRED

Shows the firmware required for the system to work correctly.

Syntax:

```
+system firmwares-required
```

Example:

```
+system firmwares-required
List of required firmwares for detected hardware
------------------------------------------------
   Filename                        Description
-------------- ---------------------------------------------------
 fw000000.bfw   Alcatel-SGS Thomson DynaMiTe ADSL over POTS
+
```

3.2.23.9 SYSTEM HEALTH

Command history:

Release	Modification
11.00.05	This command option was introduced as of version 11.00.05.
11.01.01	This command option was introduced as of version 11.01.01.

Note

This command is only available on devices designed to report this information (not all devices can do so).

Lists status information on the various equipment.

Syntax:

```
+system health ?
  fan     Print fan current speed
  psu     Print PSU voltage output value
  temp    Print temperature values
+system health
```

3.2.23.9.1 SYSTEM HEALTH FAN

Shows the current fan speed, in revolutions per minute (rpm).

Note

This command is not available on devices without fans or on devices with fans that are unable to report speed.

Example 1:

```
+system health fan
CASE1 fan speed: 5581 rpm (69 %)
```

```
CASE2 fan speed: 5714 rpm (71 %)
CASE3 fan speed: 5581 rpm (69 %)
CASE4 fan speed: 5454 rpm (68 %)
+
```

3.2.23.9.2 SYSTEM HEALTH PSU

Shows the current PSU (Power Supply Unit) voltage measurements and the expected value range within which the voltage of each PSU is considered normal.

Note

This command is not available on devices with PSUs that are unable to report voltage values.

Example 1:

```
+system health psu

PSU #1:  0.00 V (Normal values range from 11 V to 14 V)

PSU #2:  12.05 V (Normal values range from 11 V to 14 V)
```

Command history:

Release	Modification
11.00.07	The command output has changed as of version 11.00.07. The value range within which the voltage of each PSU is considered normal is shown.
11.01.05	The command output has changed as of version 11.01.05. The value range within which the voltage of each PSU is considered normal is shown.
11.01.01.60.06	The command output has changed as of version 11.01.01.60.06. The value range within which the voltage of each PSU is considered normal is shown.

3.2.23.9.3 SYSTEM HEALTH TEMP

Shows the temperature values of the various temperature probes. This command also shows the maximum value for each probe where temperatures begin to be considered abnormal.

Note

This command is not available in devices without temperature probes.

Example 1:

```
+system health temp
CPU_EXT temperature: 32 C (Normal values should never exceed 70 C)
POWER temperature: 33 C (Normal values should never exceed 70 C)
INTERNAL temperature: 27 C (Normal values should never exceed 70 C)
CPU_INT temperature: 55 C (Normal values should never exceed 95 C)
```

Command history:

Release	Modification
11.00.07	The command output has changed as of version 11.00.07. For each probe, the maximum value from which the temperature is considered abnormal is shown.
11.01.05	The command output has changed as of version 11.01.05. For each probe, the maximum value from which the temperature is considered abnormal is shown.
11.01.01.60.06	The command output has changed as of version 11.01.01.60.06. For each probe, the maximum value from which the temperature is considered abnormal is shown.

3.2.23.10 SYSTEM HTTP

Allows you to view information about the users currently connected to the Web.

Example:

```
+system http
```

```
  ID     USER        LEVEL       IP ADDRESS:PORT              LAST_ACCESS
---------------------------------------------------------------------------------------

  1      mon           5          192.168.214.170              01/01-22:30:06
  2      TEL10         10         192.168.214.170              01/01-22:30:06
```

Command history:

Release	Modification
10.08.34.05.12	This command option was introduced as of version 10.08.34.05.12.
11.00.05	This command option was introduced as of version 11.00.05.
11.01.01	This command option was introduced as of version 11.01.01.

3.2.23.11 SYSTEM LICENCE

Shows license information and performs actions in a new license.

Syntax:

```
+system licence ?
  check     Check licence file previously loaded
  create    Create a licence file
  files     Show information about licence files loaded in the equipment
+system licence
```

Command history:

Release	Modification
11.01.01	This command option was introduced as of version 11.01.01. This system is in use as of version 11.01. It is not available on earlier devices.

3.2.23.11.1 SYSTEM LICENCE CHECK

Allows a user to check a created license file to find out whether it is valid or has any unsupported parameters (for the current OS version).

Example 1:

```
+system licence check
CLI Error: Licence file not found or invalid Licence file.
CLI Error: Command error
+
```

Example 2:

```
+system licence check
Licence file OK
+
```

Example 3:

```
+system licence check
CLI Error: Error checking licence
Warning: possible errors in the configuration, at least these found:
line 4 ->  check
   CLI Error: Invalid functionality M5G.

(lines counting since last log-command-errors or transaction fail-on-error command)
CLI Error: Command error
+
```

3.2.23.11.2 SYSTEM LICENCE CREATE

Allows a user to create a license file. Once created, the file is checked to see whether it contains a valid license or has any unsupported parameters for the current OS version.

Example 1:

```
+system licence create
Input Licence File Hex Content: (End CTRL+P)
0123456789abcdff000000010000000200000000000000410000000000000000
00149a262aa66aaadd3c0a75163821c748d63af3940745aa9ec42f8afe5bfc29
cf8fc848d49c879b8052066f1a1885a2415fce84e805b9e502176266aae6cdf0
398abe8b52fb8e1c99d23ae7ac720dc3ca571aa0915fc2a20b50f0ea28d56247
163bb4a92d90909d0e32d33c4b0929994a3770acf1fc7d8eebf2f5327a998deb
3a7e26406cf92d2e11278a3403985e3aec5a8cd124d5637c7b084ef220d34365
5b4f478e3e2b81a6082ece798449d728c0808e5ceb7815a7679d7f94fedb50ac
8f51fe602ced5296a004f66f5e3fb991694d829be0867ccc9d44ba9c963ad8e1
064a794e275aa174452e8545fe0e520cc2cb77712135393fde90e2585a30a8c7
a5a44ca3456376c9bc72807bcd3d112455e42b25cbce6047bb9dae39ee2f5cdf
fcee23a3544b0e4c3f5cd252e5cb5237b4e9402c89308c6ca386d261725d4f5b
d55902ffe2294e2fa1bbbfd127a7f9746a0e07f5be4b3683ccec3c3eb8165f4b
8f3c13a3de698a16a2216c533337da02c8ebd4f0b8a66a97f6766339f7687daf
ae910a3e634b9d6c73666ed391d2f496d2fcef60b6dac30e074b9fdb0296135a
40751524e1634d17e1257567b946800d692205a88b15ba9b6d9999ea3c9634cc
32ac5a6f561704c14d3dfb98ef53c4c50e213122139841dc1d12c8597e939f75
5a1a0b80ee24d432e1510c91e52dc4cfe22099c5a039b9bda96af91dfa6b9202
9c4b023efe1ad078dc491948effeb834fe2b930d33e22800b2daf135ad316ca1
a2ecf6bdfddbb0ca04cfdc2b0c615ce4d3b45ef22910d07273a0177a47ac1f6a
9b4f3b2cd220cc727bd34fed93e80211340e6defe25504c4b07d2f0599d77551
aaeb0d1f75f45a2d2608ae683e98860444f1395ab7d65ffe99352948785f2052
30da79b2c83710ba00648ad9696801b812f9b38b80cd3f0d9c13fc1ac5289cb5
855ac70f321cff994da4f3ccf6c657b87127359ef30893e23362935ca5c9e3a6
7e4d7c4266abd133a1c5a883cd119321615e0bc2dc68f817faf5caf3570bf278
ebb8ec7caf404fd82f8c5db6241f0cbbace7906a80632d251da07b43036d144b
49475b2ed30a33619cff0eeeedb9657e4feed4d52e83cc8569bc9763b4a30df5
02c8a98cab8d5264c2ea67e25f1b5713d202d4b064e55b68d0a757b15c130865
f5ac9973792014359c5c35907375460a209940daa9bbabc016427688fb5658d9
65899cbbed70352e7227ebf94c6cb3df3e89db85bdca84d7f7717d374abd0fa0
3b9a346aebfbe1fdf56338c4f4997112e9915c4285eda6b18f9e228130ca5e46
1a3de55d0e9619eec9ebab207c09b3d64eceeaa3a29017e85237b33c00b2c761
ec74b35714e8666339de68b8a0bbe22f254625958935c0cb322c91e7f6e2205d
8a65368460ece50b951feb884a254f7bbb61bb906ff35ceecd72bf39cc7767f3
6bda37e6f3739a78bad6dbe4c9b0a4b5062020206c6963656e63650a09626173
652d6465766963652058595a0a096c6963656e63652d6e616d65205355504552
0a09636865636b0a2020206578697 40a5993419d3900333abf2fd592a265d927
File Successfully Saved
Licence file OK
+
```

Example 2:

```
+system licence create
Input Licence File Hex Content: (End CTRL+P)
0123456789abcdff000000010000000200000000000000411000000000000000
00187ca97881ca4d9abd7e14324db7817b2fc1d374191c4c4beee70c8be19deb
5725aa9e94263525e3317066d19fc70e389df32be8e717e557806391d39d0544
00901bea608be2352d7c993ed0cd15ada73d5b2d0f17740324808261ff822c44
cc5ecc09608dfbb9f1f3719df933129c4a9b2d8ebd4ad5bbf8ddb98cf77fd95d
d643fa81b1af5a8dcebc6b25f27171dde539c9c07d0388fb5298d268dd0498ee
382c2778e0989c6c1055fac7f3e9f0b7dd0a602cbd2b19d8ce47d878548744da
ddb5ebbba890c0d5920be99ce6422b1efff771e0d58e7e09779457d36fffa83a
0db39927c407592cd28401de53e87c6b8a5e174dfc5ccf503dbceca1556c29b6
02acae82650f4c033f3602419589f8286e8d1bca9ea98ccd76112294a44d804f
715a1650c1dd23ae1bede429f0996bcc044be7f6f3ed7983d5799e5f88288bc0
452d3d32e84cef7b6f49756f01cf72a354503b3d4c8405937eeca3a7330627b2
27c5f2ce4159b23de53d37acfba69e5cc5c9ba003a2e41e68af9dde526f31c63
327b1c5d2aa866fabc929d55bfdda7e07ad54545cb916a547642802e45821e90
1dd9374690fe8bf252629013c1f7b36d090a24d6deb2037ebffbb4e93848d2ae
971dad85f2c44855209d431208bca92603f0fe3716807be3406f5bdc135df0cd
e2bbf1afcb8b1a4984875d484bb69588f384acef044966f8997b161a67de6eac
ddd87acb4882162a16386deb59b51e78b9b7c9e7044346355511abba486dbed9
82cf7f652602ba1fbc1ab18c884c0ad9d4ed24ff8605e46956b7ae6cdeaf16f1
8fb08c8cb1034b6c436f2d98d151d1e1287ff809ce0b37d28205863f0dcdd241
77bce49a35a4c817c2c91230b561449de839f5cfc135deaf822bb33ee0f4dd76
f6ebb4fa92a76bd89b90749ea959404cdfdc1cd1ee48286af26874b810d52f0f
```

```
ad66164e56f2b65a9e804fd71ade72990a9d6cdbd66bac765794b249dc2559b9
d50de1a46a0f184b4b28e0c48cd62ed188a457bb6dac728cb3973f09fe1e20e4
d5f14daa6cac17c3106b495871ed508f318f1191fc79ace39eef646c31fd2c16
afdb69f4b71c1afce2038b98a8123495cbeca894ad27fd83682fc2be1e82c6de
7dc457df82d5af6964f9c7296ef7b5b9b6c2f152b16f52e3d97a7b47f2739e71
1a5b639ec0f0759d8b2edd06ad944b250d0940ad45ec4c57e10cb8752818ae3f
9f82e36c0cad869680d647674fa4254cb32df225e69e381b95b170ae4ac0221f
12d1ec0894ec795e3df85b0b262665588a8fcb43946995ecb64c8e95713c5ad4
ac12d4e1d2b4f51558efca686fdd5ff6fc041f1676972f801e5258ec625fef0d
4ac380a4f2afced23eec7971dcd1f4b46a54c9230052c851374c335f0a84d625
4c978f0fd1acb8f11771a8ee2dccf06b8abcfb4e7f2b4e900c59afdcdd078a39
ebf54439fe1d87bd6bd766aa2ade825ab86d2020206c6963656e63650a096261
73652d6465766963652058595a0a0966756e6374696f6e616c697479204d3547
0a09636865636b0a202020657869740a5836dc6b85dd442958ceb184c52430a3
File Successfully Saved
CLI Error: Error checking licence
Warning: possible errors in the configuration, at least these found:
line 4 ->  check
   CLI Error: Invalid functionality M5G.

(lines counting since last log-command-errors or transaction fail-on-error comma                    nd)
CLI Error: Command error
+
```

3.2.23.11.3 SYSTEM LICENCE FILES

Allows a user to check whether the device processed the license file properly on startup and whether the license is applied or discarded (due to errors).

Example 1:

```
+system licence create+system licence files

Information about .LIC files loaded in the system

New found files: 1
Disk Unit: A
Default file generated: OK
Loaded default: OK
Loaded new base: YES

Table of new found files

NAME              STATUS         SIZE      DATE      TIME
-----------------------------------------------------------
SAMPLE.LIC        Loaded         1152      10/12/00  00:00

+
```

Example 2:

```
+system licence files

Information about .LIC files loaded in the system

New found files: 1
Disk Unit: A
Default file generated: OK
Loaded default: OK
Loaded new base: NO

Table of new found files

NAME              STATUS         SIZE      DATE      TIME
-----------------------------------------------------------
SAMPLE.LIC        Errors         1152      10/12/00  00:00

+
```

3.2.23.12 SYSTEM LOGIN

Shows whether case-sensitive authentication is enabled in the username authentication process when the device is accessed. Then it shows a list of information about users who have accessed the Teldat console and asks whether we want to delete the login history.

Syntax:

```
+system login
```

Example:

```
+system login
Case-sensitive login: enabled

     Date        Login     Type
-------------- ---------- ------
01/03 10:41:01 root       REMOTE
01/03 10:42:05 user1      REMOTE
02/04 16:58:06            LOCAL
02/04 16:58:19 user1      REMOTE
02/04 16:59:55 user1      REMOTE
03/07 10:09:49 omateo     REMOTE
03/07 10:26:29 rsanchez   LOCAL
03/07 10:27:06 user1      REMOTE
03/07 10:30:06 root       REMOTE
03/07 10:30:09            LOCAL
03/07 10:30:16 root       REMOTE
Clean historic?(Yes/No)? No
+
```

3.2.23.13 SYSTEM LOGIN-HISTORIC

Displays a list of information about users who have accessed the console and asks whether we want to delete the login history.

Syntax:

```
+system login-historic
```

Example:

```
+system login-historic
     Date        Login     Type
-------------- ---------- ------
01/03 10:41:01 root       REMOTE
01/03 10:42:05 user1      REMOTE
02/04 16:58:06            LOCAL
02/04 16:58:19 user1      REMOTE
02/04 16:59:55 user1      REMOTE
03/07 10:09:49 omateo     REMOTE
03/07 10:26:29 rsanchez   LOCAL
03/07 10:27:06 user1      REMOTE
03/07 10:30:06 root       REMOTE
03/07 10:30:09            LOCAL
03/07 10:30:16 root       REMOTE
Clean historic?(Yes/No)? No
+
```

The meaning of the fields shown is as follows:

Date Date and time of access (month/day).

Login Name of the user who has connected to the device. This field will be empty if no users have been created on the device.

Type User access type: via telnet (REMOTE) or via the console (LOCAL).

3.2.23.14 SYSTEM MEMORY

Displays statistics about system memory.

Syntax:

```
+system memory
```

Example:

```
+system memory
 Caller  Second C. Third C.  Address  Size     Age     Hsh
-------------------------------------------------
xxxxxxx   xxxxxxxx  xxxxxxxx   xxxxxx  xxxx    xxxxx   xxxxxx

Times couldnt monitor a request X
Times couldnt match a free X
Entries created X Entries available X Entries in use x
+
```

> **Note**
>
> This command should only be performed by Teldat technical personnel.

3.2.23.15 SYSTEM PCMCIA

Allows you to access the PCMCIA interface monitoring environment at the physical layer (controller and card).

Syntax:

```
+system pcmcia <option> <parameters>
DEBUG
DUMP
```

- **<option>** specifies the action to be performed: enable/disable events [DEBUG] or dump controller and card status information [DUMP].
- **<parameters>** are the necessary parameters for the different actions available.

Example:

```
+system pcmcia dump socket
 Identification and revision= *(0x00)=0x84
 Chip information           = *(0x1f)=0x00

PCMCIA Socket -0/A- Controller Registers
----------------------------------------------
 Interface status           = *(0x01)=0x7f:  [bvd1/stschg] [bvd2/spkr] [detect]
[wrprot] [ready] [poweron]
 Power control              = *(0x02)=0x90:  [output] [resetdrv] [Vcc=5v] [Vpp off]
 Interrupts and control     = *(0x03)=0x70:  [iocard] [intr ena] [irq=0]
 Card status changes        = *(0x04)=0x00:
 Card status chng int cntrl = *(0x05)=0x09:  [bvd1/stschg] [detect] [irq=0]
 Misc control 1             = *(0x16)=0xc0:  [inpack]
 Misc control 2             = *(0x1e)=0x00:
 MemMap(0) = 0x21, 240 ns, 0xf0001000-0xf0001fff, 00000 [active] [attrib]
 MemMap(1) = 00, 0 ns, 00000-0x01fff, 00000
 MemMap(2) = 00, 0 ns, 00000-0x01fff, 00000
 MemMap(3) = 00, 0 ns, 00000-0x01fff, 00000
 MemMap(4) = 00, 0 ns, 00000-0x01fff, 00000
 IoMap  (0) = 0x09, 0 ns, 0x03f8-0x03ff [active] [0ws]
 IoMap  (1) = 00, 0 ns, 0000-0x0001
 TmrSet(0) = setup = 0, command = 0, recovery = 0
 TmrSet(1) = setup = 0, command = 0, recovery = 0
 ExtRegs   = mask 0 = 0x7f, mask 1 = 0x90, DMA ctl = 0x70 [dreq is inpack] [pullup]

+
```

For more information on how to monitor the PCMCIA interface, please refer to the following manual: *Teldat Dm757-I PCMCIA UMTS Interface.*

3.2.23.16 SYSTEM POWER-OFF-STATUS

Shows the current status of the **ignition** signal and the value programmed in the timer. If the ignition signal is disabled, it also shows how much time has elapsed since it was switched off.

Syntax:

```
+system power-off-status
```

Example 1:

```
+system power-off-status
 Ignition signal state ACTIVE
 Programmed time to power off: 60 seconds
+
```

Example 2:

```
+system power-off-status
 Ignition signal state INACTIVE
 Programmed time to power off: 60 seconds
 Time elapsed with ignition off: 10 seconds
+
```

3.2.23.17 SYSTEM PROCESS-LIST

Displays the percentage of CPU usage and stack status of each of the active processes in the system. The command displays three CPU usage percentages for the last 5 seconds, 1 minute and 5 minutes, respectively. You need to enable process monitoring before you can use this command. There are three types of processes:

(a) *Interrupts*. This is the highest priority process and is responsible for attending to the requests of the different hardware components, such as packet receipt or a change in an interface's physical layer.

(b) *High level interrupt handler*. These processes are responsible for attending to higher level (lower priority) interruptions captured by the *Interrupts* process.

(c) *Task.* These processes are responsible for the remaining tasks on the device, such as monitoring, configuration, routing protocols, etc.

Syntax:

```
+system process-list
```

Example:

```
+system process-list
Process monitoring enabled.
Stack status and CPU load for each process.
Type: I (interrupts), H (high level interrupt handler), T (task)

 Type  Name       Stack size   Status    cpu %  (5s/1m/5m)
----------------------------------------------------------
   I   Ints          4112      ok      0.16   0.16   0.17
   H   SYSTEM H      4104      ok      0.60   0.60   0.59
   H   UART-RXH      4096      ok      0.00   0.00   0.00
   T   DISC         32768      ok      0.00   0.00   0.00
   T   CMDMUTEX     16000      ok      0.00   0.00   0.00
   T   SAVE_TXT     16000      ok      0.00   0.00   0.00
   T   LAPB2DRV      2048      ok      0.00   0.00   0.00
   T   CONFIGUR      2048      ok      0.00   0.00   0.00
   T   DRIVER        2048      ok      0.02   0.02   0.02
   T   SYSTEM_M      2048      ok      0.00   0.00   0.00
   T   LAPB          4096      ok      0.00   0.00   0.00
   T   LAPB_MNG      2048      ok      0.00   0.00   0.00
   T   X25           4096      ok      0.00   0.00   0.00
   T   X25_MNG       4096      ok      0.00   0.00   0.00
   T   MOTPROT       4096      ok      0.00   0.00   0.00
   T   PROTMOT       2048      ok      0.00   0.00   0.00
   T   CONTINT       2048      ok      0.02   0.02   0.02
   T   CRYPX25       2048      ok      0.00   0.00   0.00
   H   SCC1          4096      ok      0.00   0.00   0.00
   H   SCC2          4096      ok      0.00   0.00   0.00
   H   SCC3           4096      ok       0.00   0.00   0.00
   H   SCC4           4096      ok       0.00   0.00   0.00
   T   GESTCON       32000      ok       0.05   0.05   0.56
   T   VISEVEN        8192      ok       0.00   0.00   0.00
   T   TASKER        32000      ok       0.92   0.85   0.80
   T   CONTROL        4096      ok       0.00   0.00   0.00
   T   MTC            4096      ok       0.00   0.00   0.00
   T   RESET          4096      ok       0.00   0.00   0.00
   T   BFD           32768      ok       0.01   0.01   0.01
   T   FTP           32768      ok       0.00   0.00   0.00
   T   SCEP_ACT       4096      ok       0.00   0.00   0.00
   T   SCEP_WAL       4096      ok       0.00   0.00   0.00
   T   SNMP          32768      ok       0.00   0.00   0.00
   T   SNMP-TRA      32768      ok       0.00   0.00   0.00
   T   TELNETSR       8192      ok       0.09   0.17   0.03
   T   DNSCACHE      32768      ok       0.00   0.00   0.00

+
```

The meaning of the fields shown is as follows:

Type Type of process (*I* for *Interrupts*, *H* for *High level interrupt handler*, and *T* for *Task*).

Name Name of process.

Stack size Size of the task's stack, in bytes.

Status Stack status. *Ok* means the stack is working properly. *Overflow* means the stack has overflowed due to lack of memory. *Unknown* means there are unmanaged stacks.

cpu % (5s/1m/5m) CPU usage percentages for each process. Data is shown from left to right and each column shows usage for the last 5 seconds, 1 minute and 5 minutes (respectively).

Note

Enabling process monitoring can adversely affect the performance of the device.

3.2.23.18 SYSTEM SSH

Displays information on the users that are connected to the device by SSH.

For more information, please see the following manual: *Teldat Dm787-I SSH Protocol* .

Syntax:

```
+system ssh
```

Example:

```
+system ssh

Time unit: minutes
  ID  USER   LEVEL     IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME  IDLETIME TIMEOUT
---------------------------------------------------------------------------------------


  0  admin  15          Local Console  12/10/00 02:57:23        0        1        0
  1  admin  15     192.168.213.14:46107 12/10/00 02:57:07        0        1        0 *
+
```

3.2.23.19 SYSTEM STACK-STATUS

Shows the stack status of each system process. Each system process has its own stack memory (where the current process's state is stored). This command allows you to see the status of the stack of each of the active processes in the system. There are four types of processes:

(1) *Startup*. These processes are run at the very beginning and are in charge of setting up the system, reading the configuration, and carrying out other initialization tasks. The stacks used by the startup processes are deleted later on, but the status is saved for future reference.

(2) *Interrupts*. This is the process with the highest priority and it answers the requests of different hardware components, such as the packets received or a change in an interface's physical layer.

(3) *High level interrupt handler*. These processes are responsible for attending to higher level (lower priority) interruptions captured by the *Interrupts* process.

(4) *Task* . These processes are responsible for the remaining tasks (such as monitoring, configuration, routing protocols, etc.).

Syntax:

```
+system stack-status
```

Example:

```
+system stack-status
Stack status for each process.
Type: S (startup) I (interrupts), H (high level interrupt handler), T (task)

 Type  Name       Stack size (addr)   Curr.    Min.   Status
------------------------------------------------------------
   S   Startup1    4096  (01EE2DA0)    4096    3660    ok
   S   Startup2   32000  (0214A020)   32000   23900    ok
   I   Ints        4096  (02154020)    4096    2860    ok
   H   SYSTEM H    4104  (02158020)    3856    3732    ok
   H   UART-RXH    4096  (021F2020)    3848    3792    ok
   T   DISC       32768  (021F6020)   32520   30812    ok
   T   CMDMUTEX   16000  (02201020)   15752   15624    ok
   T   SAVE_TXT   16000  (0222D020)   15752   15600    ok
   T   GESTCON    32000  (02426020)   31752   26468    ok
   T   VISEVEN     8192  (02430020)    7944    7496    ok
   T   TASKER     32000  (02435020)   31752   26648    ok
   T   CONTROL     4096  (0221B020)    3848    3816    ok
   T   MTC         4096  (0243F020)    3848    3720    ok
   T   RST-REST    8192  (02443020)    7944    7848    ok
   T   RESET       4096  (02448020)    3848    3884    ok
   T   NVRAM_EM    4096  (0244C020)    3848    3684    ok
   H   FAC-RST-    4096  (02450020)    3848    3944    ok
   T   LEDSTSK     4096  (0214E020)    3848    3604    ok
   T   events     16384  (02454020)   16136   15984    ok
   T   ELOOP_HO    8192  (02475020)    7944    5560    ok
   T   BFD        32768  (03B55020)   32520   32556    ok
   T   FTP        32768  (03B91020)   32520   32400    ok
   T   RTMNGD     32768  (03BFE020)   32520   27704    ok
   T   wifi1       4096  (03C13020)    3848    3720    ok
   T   PTHR-1     16384  (03D29020)   16136   15636    ok
   T   ath0        4096  (03D30020)    3848    3620    ok
   T   RIPNGD     32768  (03D63020)   32520   31988    ok
   T   OSPFv3     32768  (03D74020)   32520   31956    ok
   T   SCEP_WAL   16384  (03DCF020)   16136   15760    ok
   T   SNMP       32768  (03DEE020)   32520   31916    ok
   T   SNMP-TRA   32768  (03DF9020)   32520   32240    ok
   T   TELNETSR    8192  (03E04020)    7944    7228    ok
   T   SELI TSK    8192  (03E0D020)    7944    7160    ok
   T   HTTPDSM     8192  (03E12020)    7944    7880    ok
   T   HTTPDSRV    8192  (03E17020)    7944    4404    ok
   T   HTTP0      32000  (03E2E020)   31752   30848    ok
   T   TWMPM       4096  (03E38020)    3848    3884    ok
   T   HSTSK       4096  (03E3C020)    3848    3768    ok
   T   UAMMNG      8192  (03E40020)    7944    7792    ok
   T   PTHR-2     16384  (03E45020)   16136   16172    ok
   T   PTHR-3     16384  (03E4C020)   16136   16172    ok
   T   MNGPLATC   32000  (03E63020)   31752   30832    ok
   T   PTHR-4     16384  (03E6D020)   16136   16172    ok
   T   USBM       32768  (03E76020)   32520   30972    ok
   T   VRXDSLCT    8192  (03EAB020)    7944    6236    ok
   T   PTHR-5     16384  (03EB0020)   16136    7500    ok
   T   PTHR-7     16384  (040F4020)   16136   15628    ok
   T   PTHR-8     16384  (040FB020)   16136   15740    ok
   T   PTHR-42    16384  (03F2E020)   16136   14928    ok


+
```

The meaning of the fields shown is as follows:

Type Type of process (*I* for *Interrupts* , *H* for *High level interrupt handler* , and *T* for *Task*).

Name Name of process.

Stack size (addr) Size of the task's stack, in bytes. The stack's base address is shown in brackets.

Curr. Minimum number of free bytes detected in the stack in calls to the operating system.

Min. Minimum number of free bytes detected in the stack.

Status Stack Status. *Ok* means the stack is working properly. *Overflow* means the stack has overflowed due to lack of memory. *Unknown* means there are unmanaged stacks.

> **Note**
>
> This command should only be performed by Teldat technical personnel.

Command history:

Release	Modification
11.01.05	The "startup" process under the "system-stack status" command was introduced.

3.2.23.20 SYSTEM TELNET

Displays information about the users that are connected to the device.

Syntax:

```
+system telnet
```

Example:

```
+system telnet

Time unit: minutes
  ID  USER   LEVEL     IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
--------------------------------------------------------------------------------------


  0  user1    15           Local Console  03/03/05 10:40:57        0        0        0
  2  user1    15      172.24.51.128:59671 04/05/05 16:59:46        0       10        0 *
  1  user1    15       192.168.1.1:0     04/03/05 16:57:58        2        0        0
+
```

The meaning of each field shown is as follows:

ID ID number of the Telnet session.

USER Name of the user connected to the device. This field will be empty if no users have been created.

LEVEL User privilege level.

IP ADDRESS:PORT IP address and port from which the connection is received.

CONNECTION TIME Date and time the connection occurred.

INACTIVITY TIME Telnet session downtime. If this parameter is disabled in the device configuration, its value will be 0.

IDLETIME Maximum idle time allowed. If no value is specified, then the value of this parameter will be 0.

TIMEOUT Maximum session time allowed. If no value is specified, then the value of this parameter will be 0.

An asterisk next to the idle time field indicates the session from which you are accessing.

3.2.23.21 SYSTEM TELNET-CLIENTS

Displays information about open Telnet sessions between the router and the remote devices.

Syntax:

```
+system telnet-clients
```

Example:

```
+system telnet-clients
Session  Local-user     VRF       Local-IP      Remote-IP       Session-start    URL
```

```
------------------------------------------------------------------------------------
  1          tel1    <main>     172.16.0.1     172.16.0.2   10/02/12 12:15:48    --
  2          tel2    <main>     172.16.0.1    172.16.0.24   10/02/12 12:15:49    --
  3          tel3    <main>  2001:db8:1::2  2001:db8:1::1   10/02/12 12:15:50    --
  4          tel4    <main>     172.16.0.1     172.16.0.2   10/02/12 12:15:51    --
 17         tel17      vrf2     172.17.0.1     172.17.0.2   10/02/12 12:16:04    --
```

The meaning of each of the columns displayed by the command is as follows:

SESSION This is the unique ID number the router assigns to the Telnet session.

LOCAL-USER Name of the registered user that opened the Telnet session.

VRF VRF table used to reach the remote device.

LOCAL-IP IP address of the device used to communicate with the remote device.

REMOTE-IP IP address of the remote device on which the Telnet session is opened.

SESSION-START Date and time at which the session was opened.

URL URL of the remote device on which the Telnet session was opened, if it was used.

3.2.23.22 SYSTEM TERMINAL

Allows you to exchange messages and commands between the terminals of the different sessions established with the monitored device.

Syntax:

```
+system terminal <option>
  kill-terminal    Kill another terminal
  send-escape      Send an escape character to another terminal
  writeln          Write a line to another user
```

3.2.23.22.1 SYSTEM TERMINAL KILL-TERMINAL

Forces a session to close. Once this command is executed, no other data is taken into account. Instead, the user is shown a list of active sessions and asked to select the one he wants to terminate. This command is **only** available to users with ROOT privileges (access level 15).

Example:

```
+system terminal kill-terminal ?
  <cr>
+system terminal kill-terminal
Current active terminal sessions:
Local console session:
Time unit: minutes
  ID  USER   LEVEL   IP ADDRESS:PORT  CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT

  0  user1    15       Local Console    09/12/13 19:21:34        0        0        0 *
Current active Telnet sessions:
Time unit: minutes
  ID  USER  LEVEL   IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
-----------------------------------------------------------------------------------

  1  rafa   15   [192.168.212.26]:47354 09/12/13 19:22:49        0        0        0
  2  pepe   5    [192.168.212.26]:47353 09/12/13 19:22:24        0        0        0
Current active SSH sessions:

Time unit: minutes
  ID  USER  LEVEL   IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
-----------------------------------------------------------------------------------
  3  rafa   15     192.168.212.26:47621 09/12/13 19:22:10        0        0        0
Destination console ID:[0]? 1
You are going to kill the terminal in [192.168.212.26]:47354
Are you sure(Yes/No)? yes
+
```

The remote terminal displays the following:

```
Message from user1 in console ID 0:
Your session is going to finish
```

```
Connection closed by foreign host.
```

3.2.23.22.2 SYSTEM TERMINAL SEND-ESCAPE

Sends the escape character (*Ctrl+p* by default) to a particular terminal session. Once this command is executed, no other information is taken into account. Instead, the user is shown a list of active sessions and asked to select the one to which he wants to send the escape character; thus causing the user that initiated the other session to leave the console menu. This command is **only** available to users with ROOT privileges (access level 15).

Example:

```
+system terminal send-escape ?
  <cr>
+system terminal send-escape
Current active terminal sessions:
Local console session:

Time unit: minutes
  ID  USER   LEVEL   IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
------------------------------------------------------------------------------------
  0  user1    15       Local Console  09/12/13 19:21:34        0        0        0 *
Current active Telnet sessions:
Time unit: minutes
  ID  USER   LEVEL   IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
------------------------------------------------------------------------------------
  1  rafa   15    [192.168.212.26]:47357 09/12/13 19:41:30        0        0       0
  2  pepe   5     [192.168.212.26]:47353 09/12/13 19:22:24        0        0       0
Current active SSH sessions:

Time unit: minutes
  ID  USER LEVEL     IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
------------------------------------------------------------------------------------
  3  rafa   15     192.168.212.26:47621 09/12/13 19:22:10        0        0        0
Destination console ID:[0]? 1
You are going to send the escape sequence to the terminal in [192.168.212.26]:47357
Are you sure(Yes/No)? y
Escape sequence sent
+
```

The remote terminal shows the following:

```
Message from user1 in console ID 0:
I have sent you the escape sequence
```

3.2.23.22.3 SYSTEM TERMINAL WRITELN

Allows a user to send a message to another terminal (i.e., to the user of another session established with the monitored device). Once this command is executed, no other information is taken into account. Instead, the user is shown a list of active sessions and asked to select the one to which he wants to send a message (-1 to send the message to all sessions). He is then asked to type the message. This command is available to all users with MONITOR-level access or above.

Example:

```
+system terminal writeln ?
  <cr>
+system terminal writeln
Current active terminal sessions:
Local console session:
Time unit: minutes
  ID  USER   LEVEL   IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
------------------------------------------------------------------------------------
  0  user1    15       Local Console  09/12/13 19:21:34        0        0        0 *
Current active Telnet sessions:
Time unit: minutes

  ID  USER   LEVEL   IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
------------------------------------------------------------------------------------
  1  rafa   15    [192.168.212.26]:47357 09/12/13 19:41:30        0        0       0
  2  pepe   5     [192.168.212.26]:47353 09/12/13 19:22:24        0        0       0
```

```
Current active SSH sessions:

Time unit: minutes
  ID  USER  LEVEL    IP ADDRESS:PORT   CONNECTION-TIME INACTIV-TIME IDLETIME TIMEOUT
------------------------------------------------------------------------------------
  3  rafa   15     192.168.212.26:47621 09/12/13 19:22:10        0        0        0
Destination console ID (-1 for broadcast):[0]? -1
Message text (maximum 100 characters): []? I have to restart the router

Message from user1 in console ID 0:
I have to restart the router
Message sent
+
```

The following will appear in one of the remote terminals:

```
Message from user1 in console ID 0:
I have to restart the router
```

3.2.23.23 SYSTEM USB

Allows you to access the USB monitoring environment at the physical layer (controller and card).

Syntax:

```
+system usb <option> <parameters>
DEBUG
LIST
```

- **<option>** specifies the action to be performed: enables/disables events [DEBUG] or lists any available information on the various options [LIST].
- **<parameters>** values that need to be entered to carry out the available actions.

Example:

```
+system usb list supported
Option Fusion Globettroter
   Manufacturer ID 0x0af0 Card ID 0x6000
Vodafone Connect 3G
   Manufacturer ID 0x0af0 Card ID 0x5000
Option Globettroter Quad
   Manufacturer ID 0x0af0 Card ID 0x6300
Option Globettroter 3G GT Fusion Lite
   Manufacturer ID 0x0af0 Card ID 0x6100
Novatel Merlin U740 R.0 HSDPA
   Manufacturer ID 0x1410 Card ID 0x1400
Novatel Merlin U740 HSDPA
   Manufacturer ID 0x1410 Card ID 0x1410
Novatel Merlin V620 CDMA EV-DO
   Manufacturer ID 0x1410 Card ID 0x1110
Sierra Aircard 580
   Manufacturer ID 0x1199 Card ID 0x0112
Huawei Mobile Connect E612
   Manufacturer ID 0x12d1 Card ID 0x1001
+
```

3.2.24 TFTP

Accesses the device's *Trivial File Transfer Protocol* (TFTP) client.

Syntax:

```
+tftp
```

Example:

```
+tftp
TFTP manager
TFTP+
```

For more information on the *TFTP* client interface, please see the following manual: *Teldat Dm765-I TFTP Protocol.*

3.2.25 TELEPHONY

Provides access to the device's VoIP telephony monitoring environment.

Syntax:

```
+telephony
```

Example:

```
+telephony
Telephony Monitor
Telephony Mon+
```

For more information on this monitoring environment, please see manual: *Teldat Dm722-I Telephony Over IP.*

3.2.26 UCI

Allows you to configure the **Teldat Router**'s encryption unit.

Syntax:

```
+uci <option>
HELP_STATISTICS
INIT_STATISTICS
LINE_X25
RESET_LINE_X25
STATISTICS
GENERAL_CRYP
CLEAR_STACRYP
```

- **< option >** specifies the type of information to monitor.

Example:

```
+uci help_statistics
Statistics meanings
RECEIVED FRAMES REJECTED
        TOO_LARGE:      The received frame has, or has not, too large size
                        coincided with encryption header
        FAILURE:        Frame reception failure
        WITHOUT.LINE:   Frame received but impossible to be transmitted to
                        destination because the receiver was not ready
        WRONG.ENCRYPT:  Impossible to encrypt a received frame
        WITHOUT.MEM:    Not enough memory for the transmitted frame

CONTROL FRAMES RECEIVED
        DLCI not between 16 and 1007 (included)

PROCESSED FRAMES
        ENCRYPTED:      Frames encrypted correctly
        DECRYPTED:      Frames decrypted with DLCI key
        DEC.KEY.DEF:    Decrypted frames with the default key, not decrypted
                        with the DLCI key
        TRANSPARENTS:   Transparent frames

TOTAL PROCESSED FRAMES =ENCRYPTED + DECRYPTED +  DES.KEY.DEF + TRANSPARENTS
0          0          0             0             0
+
```

3.2.27 UPTIME

Displays time statistics about the device, including the current date and time and the time elapsed since the last restart.

Syntax:

```
+uptime
```

Example:

```
+uptime

Date:    Monday, 12/15/14     Time: 17:39:00
Router uptime: 8m9s

+
```

Command history:

Release	Modification
11.00.03	New command added.

3.2.28 VERSION

Displays all information relating to hardware, license, boot ROM version, software version, cellular driver and wireless LAN driver versions.

Syntax:

```
+version
```

Example:

```
+version

Teldat Router, M1 4GESW SLOT1 IPSec SNA VoIP T+ 34 12  S/N: 819/02727
Profile: none
ID: TM1-31F128R L34.12

Boot ROM release:
 BIOS CODE VERSION: 03.00  Oct 22 2014 18:12:24 L0

System Info:
PCB:0x13A GPPORCR:0x00000000 PVR:0x80212151 SVR:0x80F90110
CLKs: CCB=396000 CPU0/1=792000/0 DDR(clk)=330000 LBUS=99000 PCI0/1=0/0
Watchdog:Enabled
MMU Mode:Dynamic
ICache:ON DCache:ON Write-Back L2Cache:ON

Software release: 11.00.03-Beta-01aac0b Dec 15 2014 11:22:07
Compiled by integrator on ares.id.teldat.com
Loaded from primary partition

Cellular Driver Version: 00.09

WLAN Driver Version: 9.5.0.35.1

+
```

Command history:

Release	Modification
11.00.03	New command added.

3.2.29 WEB-PROBE

Accesses the web-probe monitoring menu.

Syntax:

```
+web-probe
```

Example:

```
+web-probe
```

```
-- Web Probe user monitoring --
PROBE+
```

3.2.30 LOG

Provides additional information on device operation. It is only useful when carrying out Teldat technical support tasks. This information is shown as an hexadecimal dump per screen.

Syntax:

```
+log <number>
```

- **< number >** number of items to save/show.

Example:

```
+log 5
0000 0000 0100 004A 0D0A 3031 2F30 312F
3030 2030 303A 3030 3A30 3020 392E 312E
3720 4D61 7220 3133 2032 3030 3220 3137
3A33 303A 3139 2062 7920 2020 6D62 6572
726F 6A6F 206F 6E20 204D 4245 5252 4F4A
4F32 007D
+
```

The output of this command varies for newer versions, where the actual log is contained between two tags. To properly read the log given, *copy the BEGIN LOG and END LOG tags and the content* itself. Here is an example of the output given:

```
+log 100

-----BEGIN LOG-----
H4sICAAAAAAAA21mbG5tLnR4dADtWW1vGzcS/m7A/4Hopxi9lYevSwr2HbhcrqNW
sYyV2hQ4HIqVtAoMKJIhyWmuv77DXa0l6yrFPqdJHZiSQA45MxwOyeFD6viIwill
pwAEoF19CXwELkCmyqmIcwD48fXvpJufA8TlaEQ18Xney9tkNB+XryYlwMnRMcH0
avlbVC4W88UJpS3Ar46SclVEYjgaFway29lodT2fEVdMp8s2MdJQcRbF3jmqq9xT
CDmyurPIYXWKOQPpHbbDv86i4z3WshicTNyWtdejcrYs24RLQvdKKSZiFW+k+qeX
baLBICfl6h8kLT+gHjIr3qMmcVHuVSSMstRuFJE/dQC5+QgMTlfldFysxLvy1+Wq
mI2LxZj8cDsjLCYMqCEU1dK2VGT4X/Tp+3fz2e/FdE7Qb8WiXLaux61aQWs0f0+6
HXd+YIgUYi9NujVEkvu+H7yyPw16eX/Qx1lENkwafyfHR+Ti6qqXu7vqkMjVzxWt
gVFGJSX9hs4MUAoolXVt/zVJOr0+cb3Uk5993u/00JkgWyAJsTcLwmE9PoGDa0tO
ukBev6WC/bntygonnLPZxnbsiJDFCcnLZbnCYu/H1h5Zk2U+80ztyE5PSHdejMlk
ga6bFrNDOoSJAZwSW4uqX3y4nr0jxZIUNzdh8dPW8Hq2T1ylFhIm/kd8dL3aLwZU
JkJxyncsJ+T8/Jx0LjuDju0S8sZf/hRqqra9Q1hPf6ye4fSDyaS1mqZbe7pYrkj5
oZytlu29QiKGhJuN0A6bUGjNoIXbkPTf9Mn0GlUub0ejcrkMO+x6tpqQUTmd3k6L
BT3Fse3tiEvIhHhIR3bg3qRR9E87+N69ueief2e73e8e0xVjwC3s7YprQi47rgXA
iA/uIbc3ZDUn4+tiWi6I6+S/9q785b/Zf8goRN5VsSJmt38a+q88vPiIoXn8sI8c
hRWCi2CdJuuEgXqylZCFxZO4lBMxDktNSORlKCwmrBajdCjMZFcNcgr8cWBCSzWh
ED5qzTRWZbNyP52w/+OjwaLAaF5+aJN+jlOjzVmU5Bin1Vl08RanitU5xfq0N6C/
oEfpveLDuK56XdQK/CzqXGEen0WXAyxwui4w0xR0U2iagNVC6sGdNVzN6CazNikL
XnI8ScuClmZ0FhV0OByKkI/KMdJU8phpUReo1g/hGMlYlqhTCxAs6DQTw6DKYRQ0
mHI8DHlBQYT6AiZjUfHj7P0/NrBSFUrvPfONt0rpdPvQ9b84f9Umb3HDvU57F1XY
JK+qiLQOZ2kV5Vw/z+E8VFlheSZIqKBVBU4OrsgK7NTN1OLO3EUuVpgsRVOtAK/q
PI3RVKtSSOpcaYQulhpvsN1bwTRUubR1rpKqXoJFPqWF8BBy5YK8MgZNrXLlRKWH
+goSaUurPFh5HxrlAFG91uOM5EAjoFlK4xSQYNFdMM+BRxLDCU8zi0dLDqIhPbbJ
bUYVZeuERLxpCVJ6m9FEUtOMc25JTmGrhdI1wXyQomgHTfS6jUcGLDqEMyTEthRa
ETsVOxYItd0SbxNoAwNcOxWbqVqo05V9DCoVslLBgi/QhRgqkNj2BeMNQSupOyvQ
JCaDestVpWLbFywOyrUKsT1I6UAmuC0CI9rBuWfCIcGhVmFCC6dVC8W1QvpX3+My
/tjM0PFRVQpbBeql1VgFd5H1cEINn0jNnDeTXdfGOowk817UGjbBNMvw523Mm6qN
JTQxtZNq7zQacFBGNaPSPNgUOyMzts1ZuanxXs2pqlGHjVTNukA3xJYlnm9GJe80
NHSjqdJgag24BYFlMuXKret149m45jRSNy077qlVM2V3e9wH+bmymdXykTcOLhOb
Cv30GwcXjHuv9t44cLErPSkO3zjwJldBMsbawNc3DoQF5btFsZovnnDjiBP+LCFn
IqX0VD8OciZCmAT0Xw85Ey7xbH1QR0+FnAn19t7V6xuCnOO4FguQc1jsqtmBnGwH
co7vR8lD6VOQ89lBzIarXmOHIefzgphsEsdl3S7NAchpLR6eTD8Zcqb6BXI+FXIm
IBmPs+RrQM7kBXK+QM4vDjkR8Tkm6WMfuTk6VbvP8MhNrRKJ/JtCTmv1s4ScmfTM
muSxr5z6/kr46145RZxa+UVeOWlm8VD6FiEnn/BajAZ8Ue6q2YGcfAdyju5HyUPp
5ZXzG3zlREQkidDTvcCOR8EOe+IPZATlJdfA3L6F8j5Ajm/PORMhfX+sa+czDHh
ffYZIKdNuTfZ3xRyxql7lpAzkSpJJHvsKydLM3cAcsrP9soJLE72Pz1udfT0V06X
Mf5tQs5y88e6LD71x7rYgZzD+1HyUPo8kPOrvmdW9N4H2T8AADF0KyEnAAA=
-----END LOG-----
+
```

Command history:

Release	**Modification**
11.01.08	The *"log"* command's output format changed as of version 11.01.08.

Chapter 4 Event Logging System ELS

4.1 Introduction

This chapter describes the Event Logging System (**ELS**). It also describes the VISEVEN process and how to get messages from the Event Logging System. The VISEVEN process provides information on the internal operation of the device and its interfaces.

The chapter is divided into the following sections:

(a) Event Logging System.

(b) Event Logging System user interface.

(c) Event Logging System commands.

4.2 Event Logging System

Events happen continuously while the device is operating. They can be caused by any of the following reasons:

- System activity.
- State changes.
- Service requests.
- Transmission and reception of data.
- Errors in internal system data.

The Event Logging System (ELS) is a monitoring mechanism that generates messages as a result of router activity. When something happens, the ELS receives system data that identifies the source and nature of the event. A message is then generated whereby the data received forms part of the message.

By using ELS commands properly, you can sort out which messages you feel are important to the user and then display them, send them as traps or through syslog messages.

The Event Logging System and the MONITOR process counters allow you to isolate problems in the device. A quick scan of the messages will tell you whether the device has a problem and where to start looking if it does.

Commands entered at the *ELS config>* prompt create a default configuration that takes effect after you restart the device.

Commands entered at the *ELS config$* prompt create a configuration that takes effect immediately without having to restart the device.

Occasionally, you may want to obtain messages using different parameters to the ones originally set up in the ELS configuration process (*ELS config>* or *ELS config$)*. You can do this in the ELS monitoring process (*ELS+* prompt), without having to restart the router. The commands at this prompt allow you to temporarily change the selected events to be displayed on screen or sent as traps or through syslog messages. These changes take effect immediately, and are not stored in the system configuration.

Running alongside the ELS, there is another system that stores logs in non-volatile memory. These logs record information about system access (by ftp or telnet), restarts, configuration changes, and so on. Unlike events, they are stored in non-volatile memory, which means that they are stored even if the device restarts the application or shuts down.

The process for accessing the *ELS Config>* prompt from the Config> prompt, the *ELS config$* prompt from the Config$ prompt, and the *ELS+* prompt from the **+** prompt, is summarized below:

Event Logging System Configuration

To enter the ELS configuration process:

(1) At the management console prompt (*****), type the **status** command to get the configuration process identifier (PID).

```
*STATUS
System Processes:
 PID  NAME

  1   Main console
  2   Event viewer
  3   Monitor console
  4   Config console
  5   Running config console
  6   Telnet client
*
```

(2) Type **process** and the PID, in this case number 4, to enter the CONFIG process.

```
*PROCESS 4
Config>
```

You can also access the configuration environment using the **config** command at the management console prompt (*****).

```
*CONFIG
Config>
```

(3) Type **event** to access ELS.

```
Config>event
-- ELS Config --
ELS config>
```

Now you can use the ELS commands.
To leave the ELS configuration process, type **exit** to return to the Config> prompt.

```
ELS config>exit
Config>
```

> **Note**
>
> If the configuration was stored in the flash memory or a smart card, all changes made in this process only take effect when the device is restarted.

If we want the updated settings to take effect dynamically (i.e., without having to restart the device), then we need to access the Event Logging System's dynamic configuration environment.

(1) From the management console prompt (*****), type **process** and the PID (in this case 5) to enter the dynamic configuration process.

```
*PROCESS 5
Config$
```

You can also access the dynamic configuration process by typing the **running-config** command at the management console prompt (*****):

```
*RUNNING-CONFIG
Config$
```

(2) Type **event** to access the ELS.

```
Config$event
-- ELS Config --
ELS config$
```

Now you can use the ELS commands.
To leave the ELS dynamic configuration process and return to the Config> prompt, type **exit**.

```
ELS config$exit
Config$
```

> **Note**
>
> All changes made in this process take effect immediately. As with the static configuration process (config>), you need to save the changes in the flash memory or in a smart card to keep them after the next restart.

Event Logging System Monitoring

To enter the ELS monitoring process:

(1) Type **status** to access the monitoring process prompt (+).

```
*STATUS
System Processes:
 PID  NAME

  1   Main console
  2   Event viewer
  3   Monitor console
  4   Config console
  5   Running config console
  6   Telnet client
*
```

(2) Type **PROCESS** and the PID, in this case 3, to enter the monitoring process.

```
PROCESS 3
Console operator
+
```

You can also enter the monitoring process using the **MONITOR** command at the management console prompt (*****).

```
*MONITOR
Console Operator
+
```

(3) Type **event** to access the ELS.

```
+event
-- ELS Monitor --
ELS+
```

Now you can use the ELS monitoring commands.
To leave ELS monitoring and return to the monitoring process prompt (+), type **exit**.

```
ELS+EXIT
+
```

Viewing Events (Traces)

The VISEVEN and active processes allow you to view events that occur while the device is running (provided you enable tracing in the console).

An advantage of using VISEVEN to view events is that they are only displayed when the user needs to see them and not as they occur. By viewing them from the active process, you see them as they occur and you can also run other active process commands (meaning you can perform additional tasks or consult other information).

A disadvantage of using VISEVEN to view events is that you cannot perform any other tasks, and that events can get easily lost if the event buffer is small. Viewing them from the active process may interfere with the information shown by the process in progress and makes accomplishing tasks difficult.

To enter the VISEVEN process from the management console prompt (*), follow these steps:

(1) If you do not know the VISEVEN process identifier (PID), type **status** at the management console prompt (*****).

```
*STATUS
System Processes:
 PID  NAME

  1   Main console
  2   Event viewer
  3   Monitor console
  4   Config console
  5   Running config console
  6   Telnet client
*
```

(2) At the management console prompt (*****), type **process** and the PID (in this case 2) to enter the VISEVEN process.

```
*PROCESS 2
```

The VISEVEN process does not present a prompt or allow commands to be executed. However, it does show the messages that have been saved.

To leave the VISEVEN process and return to the management console prompt (*****), type **Ctrl + p**.

If you want to ignore all the events stored so far without displaying them, use the **flush** command.

The **hide** and **view** commands are available when viewing events from the active process. These commands must be written in full in order for them to take effect. If events are displayed in the active process, you will not be able to enter the VISEVEN process. The **view** command allows you to view them, whereas the **hide** command allows you to hide them.

Understanding Event Logging System messages

An ELS message looks like this if you type the command:

```
ELS+LIST SUBSYSTEM GW
GW.019                                C-INFO             Slf tst ifc %s
```

(Subsystem Event Number) (Type of Event) (Message Text)

Subsystem

Subsystem is a predefined abbreviated name for a router component or functionality, such as a protocol or interface. The letters GW (which stand for Gateway) identify the subsystem through which this event occurs.

Other examples of subsystem may be ARP, IP, ETH. By running the **list subsystem** command, you can obtain a list of the subsystems available on your router. This command is available in both the configuration (CONFIG or P4) and monitoring processes (MONITOR or P3).

Enter the subsystem as a parameter to an ELS command when you want the command to affect the entire subsystem. For example, the **enable subsystem GW** command causes all GW subsystem events to be picked up by the VISEVEN process when they occur.

Event Number

The *Event number* is a predefined, unique, arbitrary number assigned to each message within a subsystem. It does not indicate message priority. For example, in GW.019, 19 indicates the event number in the GW subsystem. By using the **list subsystem <subsystem name>** command, you can obtain a list of all the events in a subsystem. This command is available in both the configuration (CONFIG or P 4) and monitoring processes (MONITOR or P 3).

The event number always appears next to the subsystem abbreviation separated by a period, e.g., GW.019. The subsystem and event number together identify an individual event. They are typed as a parameter in certain ELS commands. When you want a command to affect a specific event only, type the subsystem and event number as a parameter for the command.

Type of Event

Event type or Filter Level is a predefined identifier that classifies each message according to the type of event that generates it. This identifier appears when the **list subsystem <name_subsystem>** command is executed.

TYPE OF EVENTS LIST

Identifier	Description
ALWAYS	Each time the device software is loaded, it displays copyright information and configuration confirmation.
UI-ERROR	Unusual internal errors.
CI-ERROR	Common internal errors.
UE-ERROR	Unusual external errors.
CE-ERROR	Common external errors.
ERROR	Includes all of the above error types.
U-INFO	Unusual informational comment.
C-INFO	Common informational comment.
INFO	Includes all of the above comment types.
STANDARD	Includes all error types and all comment types (default).
P-TRACE	Packet trace.
U-TRACE	Unusual operation trace message.
C-TRACE	Common operation trace message.
TRACE	Includes all of the above trace types.
ALL	Includes all events.

In this table, ERROR, INFO, TRACE, STANDARD and ALL are the result of combining different filter levels. STANDARD is the recommended filter level by default.

Groups

Groups are collections of user-defined events that are given a group name. You can enter the group name as a parameter to the ELS command. There are no predefined groups. A group must be created before its name can be specified on the command line.

To create a group, use the **add** configuration command, specify the name you want to give the group, then specify the events you want to include in the group. The events that are added to the group can be from different subsystems and have different filter levels.

Example:

```
ELS config>add ?
  <1..7 chars>    Group name
ELS config>add MYGROUP ?
  <1..11 chars>    Event
ELS config>add MYGROUP GW.019 ?
  <cr>
ELS config>add MYGROUP GW.019
ELS config>add MYGROUP PPP.001
ELS config>add MYGROUP PPP.002
ELS config>
```

After you create a group, you can use it to manage group events globally. For example, to enable the on-screen display of event messages for all events that have been added to a group named MYGROUP, include the group name on the command line, as follows:

```
ELS config>ENABLE TRACE GROUP MYGROUP
```

To delete a group, run the **delete** command.

Example:

```
ELS config>delete ?
  <1..7 chars>    Group name
ELS config>delete MYGROUP ?
  all              The whole group
  <1..11 chars>    Event
ELS config>delete MYGROUP all ?
  <cr>
ELS config>delete MYGROUP all
ELS config>
```

4.3 Event Logging System user interface

To work with the Event Logging System (ELS) effectively:

- You need to know what you want the system to analyze. Clearly define the problem or events that you want to view before using the VISEVEN process.
- Run the **clear** command in the configuration process to delete all enabled events and existing groups from the configuration, or, use the same command in the monitoring process to delete all enabled events running.
- Enable only those messages that are related to the problem that you want to identify.
- If working remotely, enable the events you think will help identify the problem to be sent as traps or through syslog messages, or analyze device behavior in the specific situation that you want to study in depth.

When enabling events to be displayed as console traces, if these messages occur too frequently and are not displayed on the screen as they occur in the VISEVEN process, the circular message buffer may become full and the initial messages can get lost. In the case of sending events as traps, if the storage buffer becomes saturated, the last traps that have not been transmitted are lost. The same thing happens with syslog messages: the oldest are preserved and the newest discarded.

You can enable/disable the messages as you receive them based on the events you are interested in.

Console Traces

You can enable tracing to the console (visible from the VISEVEN process (P2), or from the active process using the **view** command) for any individual event, group of events or subsystem. By using the **hide** command in the active process, the traces will no longer be visible.

```
ELS config>enable trace event icmp.002
```

```
ELS config>enable trace subsystem ip all
```

```
ELS config>enable trace group MYGROUP
```

SNMP Traps

ELS can be configured so that event messages are sent to an SNMP management workstation in a company-specific trap. These traps send the actual message that would be shown on screen if the event was enabled for tracing. A trap will be generated each time the selected event (enabled as an SNMP trap) occurs. For more information about configuring SNMP, please see the following manual: *Teldat Dm712-I SNMP Agent*

You can enable any individual event/group/subsystem to be sent as an SNMP trap.

For example, to enable the SNMP.002 event to be sent as a company-specific trap:

(1) At the ELS config>, ELS config$ or ELS+ prompts, enter:
ENABLE SNMP-TRAP EVENT SNMP.002

> **Note**
>
> If you are at the ELS Config> prompt, the settings must be saved and the router restarted for the changes to take effect.

(2) At the SNMP config> prompt, enter:
COMMUNITY <community> ACCESS TRAP-ONLY
HOST <ip address of the SNMP remote manager station> TRAP VERSION <v1/v2c> <community> ALL

> **Note**
>
> Settings must be saved and the device restarted for changes to take effect.

Syslog Messages

You can configure the Event Logging System to send a particular event in syslog message format to one or more remote stations. Any event, regardless of whether it is individual or belongs to a group or subsystem, can be enabled for this purpose.

For example, to enable the ICMP.002 event to be sent as a syslog message,

(1) At the ELS config>, ELS config$ or ELS+ prompts, enter:
ENABLE SYSLOG EVENT ICMP.002

> **Note**
>
> If you are in ELS Config>, the settings must be saved and the device restarted for the changes to take effect.

(2) At the SYSLOG config> prompt, configure the IP address or domain name of the server(s) that is going to be sent the notifications. For detailed information on the configuration parameters available with regard to this functionality, please see the following manual: *Teldat Dm753-I Syslog Client.*

> **Note**
>
> Settings must be saved and the device restarted for changes to take effect.

If you ping the router from any system under these conditions, the message is received in the configured syslog server.

Using the Event Logging System (ELS) to troubleshoot problems

When using the ELS to solve a particular problem, you need to enable the events related to said problem so that they are shown in the console. For example, if you know or think the problem is related to the IP protocol, enable all IP subsystem events by typing:

```
ELS+ENABLE TRACE SUBSYSTEM IP ALL
```

Once you are familiar with the different messages that appear, you can enable/disable those events that contain the information you want.

The Event Logging System allows you to specify which messages are to be displayed temporarily or permanently.

The Event Logging System configuration commands allow you to design a permanent message filter that takes effect every time the system is turned on or reset.

Monitoring commands let you apply temporary filters that ignore a permanent filter. When the system is restarted or reset, the system software removes this temporary filter.

Below are several examples of the Event Logging System:

Example 1. Starting the device

```
 *PROCESS 2                         calls the events viewing system
12/29/06 13:07:41  GW.001 System restarted - - ATLAS50 router cold start
12/29/06 13:07:41  GW.002 Portable CGW ATLAS50 Rel 10.7.0 strtd
12/29/06 13:07:41  GW.005 Bffrs: 1762 avail 1762 idle   fair 195 low 352
12/29/06 13:07:42  USB registered new driver hub
12/29/06 13:07:42  USB registered new driver serial
               click on <Ctrl + p> exiting the events viewing system
*
```

Example 2. Enabling the Ethernet interface test event

```
ELS+ENABLE ALL EVENT ETH.045
ELS+                                                          click on <Ctrl + p>
*PROCESS 2
12/29/06 13:18:05  ETH.045 Eth self-test Operational test fld 0000 ifc ethernet0/1
12/29/06 13:18:08  ETH.045 Eth self-test Operational test fld 0000 ifc ethernet0/1
12/29/06 13:18:11  ETH.045 Eth self-test Operational test fld 0000 ifc ethernet0/1
```

Example 3. GW protocol operation messages

```
ELS+ENABLE ALL SUBSYSTEM GW ALL
ELS+ click on <Ctrl + p>
*PROCESS 2
12/29/06 13:21:10  GW.026 Mnt ifc ethernet0/0
12/29/06 13:21:11  GW.022 Nt fld slf tst ifc ethernet0/1
12/29/06 13:21:12  GW.019 Slf tst ifc ethernet0/1
12/29/06 13:21:14  GW.026 Mnt ifc x25-node
12/29/06 13:21:14  GW.022 Nt fld slf tst ifc ethernet0/1
```

Conditional Events by Access List

This feature allows all traces belonging to subsystems associated with the reception and processing of packets that meet certain IP access list criteria to be shown in the console, while discarding those that don't. This way, we can follow the path a packet takes from the moment it enters the system until it leaves, identify the subsystems through which it has passed, and locate possible errors.

To enable this functionality, you must:

- Configure access to the generic access list configuration environment. For more information about configuring generic access lists, please see the following manual: *Teldat Dm752-I Access Control.*
- Add the lists in which you want to enable inbound packet tagging so that all events associated with packet processing are displayed in the console. To do this, use the **enable trace condition access-list <1....1999>** command.
- Enable all events to be displayed as console traces. To do this, use the **enable trace all** command from the configuration (CONFIG or P 4/ RUNNING-CONFIG or P 5) or monitoring process (MONITOR or P 3). This command should only be used with a corresponding filter to only show those events associated with a specific packet. Otherwise, the device shows all available events and consequently impacts the performance.

Example:

You are connected to the device via Telnet but you're not interested in seeing any telnet-related events. You can do this by configuring an access list to exclude port 23 tcp:

```
feature access-lists
; -- Access Lists user configuration --
      access-list 100
         entry 1 default
         entry 1 deny
         entry 1 source port-range 23 23
         entry 1 protocol tcp
;
         entry 2 default
         entry 2 deny
         entry 2 destination port-range 23 23
         entry 2 protocol tcp
;
         entry 3 default
         entry 3 permit
;
      exit
;
      access-list 5000
         entry 1 default
         entry 1 permit
;
      exit
;
exit
event
; -- ELS Config --
      enable trace condition access-list 100
exit
;
```

Conditional Events by interface

This feature allows all traces belonging to subsystems associated with the reception and processing of packets that arrive through a specific interface to be shown in the console, while discarding those that don't. This way, we can follow the path a packet takes from the moment it enters the system until it leaves, identify the subsystems through which it has passed, and locate possible errors.

To enable this feature, complete the following steps:

- Add the interfaces in which you want to enable inbound packet tagging so that all events associated with packet processing are displayed in the console. To do this, use the **enable trace condition interface <interface name>** command. Please note that you can add any of the router's interfaces to the list, but it only makes sense to select interfaces with an associated physical layer (i.e., base interfaces). You can also specify certain conditions, for example, that the packet must come from the global free buffers list (usually locally sourced packets), or that it be processed by a particular protocol using the **enable trace conditional global-buffers** and **enable trace condition protocol <protocol name>** commands, respectively.
- Enable all events to be displayed as console traces. To do this, use the **enable trace all** command from the configuration (CONFIG or P 4 / RUNNING-CONFIG or P 5) or monitoring process (MONITOR or P 3). This command should only be used with a corresponding filter so that it only shows events associated with a specific packet. Otherwise, the device shows all available events and consequently impacts the performance.

The following example shows you the screen output when this feature is enabled on an ethernet0/0 interface that receives a packet encapsulated in IPSec:

```
01/25/07 09:45:02  POLR.004 dis int ethernet0/0
01/25/07 09:45:02  SNAT.004 NAT_OUT: (172.25.6.0, 172.24.100.129)-> no nat
01/25/07 09:45:02  IPSEC.001 esp encode, spi f5b73944
01/25/07 09:45:02  IPSEC.031 prot 17 (172.25.6.0[5060])->(172.24.100.129[5060]) len 444
01/25/07 09:45:02  IPSEC.003 Pack ESP suc encap (80.36.189.123->83.55.22.247) spi f5b73944
01/25/07 09:45:02  IP.061 add lcl pkt to ip op q
01/25/07 09:45:02  IP.007 80.36.189.123 -> 83.55.22.247
01/25/07 09:45:02  SNAT.003 NAT_IN: (80.36.189.123, 83.55.22.247)-> no nat
```

Important

If several conditional events are enabled at the same time, any of the conditions can cause a trace to be generated.

4.4 Event Logging System Commands

This section describes the Event Logging System (ELS) commands. Each command includes a description, syntax and an example. Some commands are executed in the configuration process at the ELS config> or *ELS config$* prompt and others in the monitoring process at the ELS+ prompt.

You can also configure the Event Logging System from the dynamic configuration process (at the *ELS config$* prompt), which saves you from having to start the router again. Always remember to save the configuration if you want to keep the changes after a reboot. To do this, enter the **save** command at the *Config$* prompt.

4.4.1 Configuration Process Commands

These commands are executed in the configuration process at the *ELS config>* prompt. For the changes made in the Event Logging system (ELS) to take effect, you need to:

(1) Save the modified configuration (in the flash memory or a smart card) by typing the **save** command at the *Config>* prompt.

(2) Restart the device.

Command	Function
? (HELP)	Lists the ELS configuration commands.
ADD	Adds an event to a given group or creates a new group.
APPLY-FILTER	Causes the event filter settings to be dynamically applied to events generated at that time.
CLEAR	Clears all event and group settings from the ELS.
CONSOLE	Accesses the console's (CNSL) event menu.
DELETE	Deletes an event from a given group or deletes the entire group.
DISABLE	Disables event message display on the console screen and disables event filtering.
ENABLE	Enables event message display on the console screen and enables event filtering.
ENVIRONMENT-MONITOR	Enables the monitoring of the temperature probes, fans and PSUs.
EV-BUFFER	Changes the size of the event buffer.
FILTER	Sets rules to filter events so that only events that are currently relevant are displayed.
LIST	Displays information about enabled events, messages, and the minimum logging level for stored logs.
NO	Deletes an entry from the list of event filters.
NUMBER-EVENTS-LOG	Sets the number of events that are logged in NVRAM when a device RESET occurs.
NVRAM-LOG-PRIORITY	Sets the minimum logging level to store logs in non-volatile memory.
OPTIONS	Modifies the behavior of the Event Logging System.
PPP	Accesses the PPP events menu.
PRINT	Configures additional information to display on each event.
STOP-TRACES	Stops storing traces.
TRACE-LEVEL	Sets the level of the trace to display.
VRF	Accesses the event filtering menu through VRF.
EXIT	Exits the ELS configuration environment.

4.4.1.1 ? (HELP)

Displays all the commands available for this prompt. You can also type a question mark (?) after a specific command to list its options.

Syntax:

```
ELS config>?
```

Example 1:

```
ELS config>?
  add                     Adds an event to a specific group or creates a new
                          group
  apply-filter            Applies dynamically the events filtering configuration
  clear                   Erases all the event and group configuration from the
                          ELS
  console                 Enters the specific Console (CNSL) events menu
  delete                  Deletes an event from a specific group or the whole
                          group
  disable                 Disables events
  enable                  Enables events
  environment-monitor     Enables environment monitor
  ev-buffer               Sets the events buffer size
  filter                  Adds a filter
  list                    List configuration
  no                      Negates a command or sets its defaults
  number-events-log       Number of events to be logged in case of fatal error
  nvram-log-priority      Sets the priority of logs saved
  options                 Modify els subsystem behavior
  ppp                     Enters the specific PPP events menu
  print                   Configure els show options
  stop-traces             Stops saving traces
  trace-level             configures subsystem trace level
  vrf                     Enters the specific VRF filter events menu
  exit
ELS config>
```

Example 2:

```
ELS config>list ?
  all                     Lists the configuration and all the subsystems
  configuration           Lists the status of the subsystems, groups and events
  ev-buffer               Lists the events buffer parameters
  event                   Lists the filter level and the specified event message
  filter                  Lists status of the filtering and the configured
                          filters
  groups                  Lists the groups defined by the user and their content
  nvram-log-priority      Lists the minimum priority of logs saved
  subsystem               Lists all the events of a specified subsystem
ELS config>
```

4.4.1.2 ADD group

Adds an individual event to an existing group or creates a new group. Group names must be composed of alphabetical characters. Numbers and other types of ASCII characters are not allowed. Names can be no more than 7 characters long. You can create a maximum of 10 groups with up to 20 events per group.

Syntax:

```
ELS config>add <nom_group> <subsystem.num_event>
```

Example:

```
ELS config>add
CLI Error: Incomplete command
ELS config>add ?
  <1..7 chars>    Group name
ELS config>add MYGROUP ?
  <1..11 chars>    Event
ELS config>add MYGROUP IP.001 ?
  <cr>generados en ese momento
ELS config>add MYGROUP IP.001
ELS config>
```

4.4.1.3 APPLY-FILTER

Dynamically applies the event filter settings to the events generated at the time.

Syntax:

```
ELS config>apply-filter
```

4.4.1.4 CLEAR configuration

Clears all event and group settings form the Event Logging System (ELS).

All existing groups, events and subsystems enabled during configuration are deleted. Run this command, followed by the **save** command at the *Config>* prompt, to clear the configuration from the flash memory or the smart card.

Syntax:

```
ELS config>clear
```

Example:

```
ELS config>clear ?
  <cr>
ELS config>clear
ELS config>
```

4.4.1.5 CONSOLE

Opens the console's event menu.

Example:

```
ELS config>console
--  Console Events Configuration  --
Console Events config>
```

The console's event menu allows you to configure several parameters related to this type of events.

Syntax:

```
Console Events config> ?
  log          Includes additional information into console events messages
  no           Negates a command or set its defaults
  wait-time    Sets time to wait for console events processing before effective
               execution of a command
  exit
Console Events config>
```

4.4.1.5.1 LOG <info>

Allows you to include additional information in the messages linked to console events.

Syntax:

```
Console Events config>log <info>
  prompt       Includes command prompt into console events messages
  source-ip    Includes ip address and port from user equipment into console
               events messages
```

- **< info >** is the type of additional information to include in the event.

LOG PROMPT

Syntax:

```
Console Events config>log prompt
```

Example:

```
Console Events config>log prompt
Console Events config>
```

By default, the prompt is not included within console events.

LOG SOURCE-IP

Allows the device IP address and port used by the user to access the router through Telnet to be inserted in the user information field (usr) of the console event. If user access is granted via the local console, the text *Local Console* will appear next to the username.

Syntax:

```
Console Events config>log source-ip
```

Example:

```
Console Events config>log source-ip
Console Events config>
```

The syslog server receives this type of messages:

```
Jun 13 16:28:55 172.24.73.22 172.24.73.22 CNSL:001 usr rober
                       (Local Console)
                         exe IP config>show menu
                        Jun 13 16:29:02 172.24.73.22 172.24.73.22 CNSL:001 usr edu
                        (172.24.51.128:55)
                         exe +config
```

This option is disabled by default.

4.4.1.5.2 WAIT-TIME <time>

Sets the time that the device waits before executing a command in order to give the system time to fully process all the associated console events (including the sending of syslog messages or snmp traces).

This delay in execution only applies in the following cases:

- When running a command within the active configuration's editing process, with the exception of the **show menu**, **show config**, **show all-config** and **Ctrl+p** (escape character sequence to return to the management console) commands.
- When running the **view** command used to view the ELS messages.

Syntax:

```
Console Events config>wait-time <time>
  <0..1000>     Wait time value in 1/10 secs.
```

Example:

```
Console Events config>wait-time 5
Console Events config>
```

By default, the timeout value is 1 tenth of a second.

4.4.1.5.3 NO

Allows you to set the configuration parameters of the console event subsystem to their default values.

Syntax:

```
Console Events config>no ?
  log           Includes additional information into console events messages
  wait-time     Sets time to wait for console events processing before effective
                execution of a command
```

NO LOG <info>

Stops additional information (*prompt* or *source-ip*) from being included in the console event messages.

Syntax:

```
Console Events config>no log <info>
  prompt        Includes command prompt into console events messages
  source-ip     Includes ip address and port from user equipment into console events
                messages
```

- **< info >** is the type of additional information to exclude from the event.

Example:

```
Console Events config>no log prompt
Console Events config>no log source-ip
Console Events config>
```

NO WAIT-TIME

Sets the time (1 tenth of a second) the device waits by default before executing a command. This gives the system time to fully process the associated console events (including the sending of syslog messages or snmp traces).

Example:

```
Console Events config>no wait-time
Console Events config>
```

4.4.1.6 DELETE group

Deletes an event from an existing group or deletes the entire group. If the specified event is the group's last event, a message will appear. If you specify **all** instead of **subsystem.event_num**, then the entire group is deleted.

Syntax:

```
ELS config>delete <nom_group> <subsystem.event_num>
```

Example 1:

```
ELS config>delete ?
  <1..7 chars>    Group name
ELS config>delete MYGROUP ?
  all              The whole group
  <1..11 chars>    Event
ELS config>delete MYGROUP IP.001
ELS config>
```

Example 2:

```
ELS config>delete MYGROUP all
ELS config>
```

4.4.1.7 DISABLE

Selects and disables events so their messages are not displayed on screen or sent out as traps or via syslog messages. Groups, subsystems and all traces (provided you enable them beforehand with the **enable trace all** command) can be disabled. It also allows you to disable event filtering.

If you have selected an input interface to display all the traces associated with the processing and path of the packets received on that interface, you can use the **disable trace condition interface <interface name>** command to disable packet tagging on that interface. You proceed in the same way to disable the global buffer and protocol conditions with the **disable trace condition global-buffers** and **disable trace condition protocol <protocol name>** commands, respectively.

Syntax:

```
ELS config>disable
ALL
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
FILTER
SNMP-TRAP
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SYSLOG
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
TRACE
       ALL
       CONDITION INTERFACE <nom_interfaz>
       CONDITION GLOBAL-BUFFERS
       CONDITION PROTOCOL <nom_protocolo>
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
```

Example:

```
ELS config>disable TRACE EVENT ICMP.001
ELS config>
```

This example disables a single event (CMP.001) so that it is not displayed on screen.

Example:

```
ELS config>disable SYSLOG GROUP  MYGROUP
ELS config>
```

This example disables MYGROUP events so that they are not sent as syslog messages.

Example:

```
ELS config>disable ALL SUBSYSTEM IP INFO
ELS config>
```

This example disables IP subsystem INFO level events so that they are not displayed on screen or sent as SNMP traps or syslog messages.

4.4.1.8 ENABLE

Selects and enables events so their messages are displayed on screen, sent out as traps or transmitted via syslog messages. Groups and subsystems can be enabled. It also allows you to enable event filtering.

Under the present chapter's *Event Logging System user interface* on page 158 section, we explained how to activate the conditional events functionality. In brief: you use the **enable trace condition interface <interface name>** command to enable packet tagging for a given interface so that the events associated with the processing of packets received on that interface are displayed as console traces.You can also filter events by IP access list using the **enable trace condition access-list <list>** command.

We also indicated that you can enable all available events to be displayed as traces using the **enable trace all** command. We do not recommend using this command unless the conditional event functionality is used to restrict the number of traces shown to only one packet. Otherwise, the device shows all of them and consequently impacts performance.

In the same section we indicated that you can specify a trigger condition to enable tracing when the free global buffers are used (usually locally sourced packets) or when a packet is processed by a particular protocol, using the **enable trace condition global-buffers** and **enable trace condition protocol <protocol name>** commands, respectively.

If you want to debug the conditional event functionality, you can enable a special trace using the enable condition-debug command, which is printed whenever a packet is unmarked.

Syntax:

```
ELS config>enable
ALL
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
CONDITION-DEBUG
FILTER
SNMP-TRAP
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP-GROUP1
       EVENT <subsystem.event_num>
       GROUPS <nom_grupo>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP-GROUP2
       EVENT <subsistema.num_evento>
       GROUPS <nom_grupo>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP-GROUP3
       EVENT <subsystem.event_num>
       GROUPS <nom_grupo>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP-GROUP4
       EVENT <subsystem.event_num>
       GROUPS <nom_grupo>
       SUBSYSTEM <subsystem> <filtered_layer>
SYSLOG
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
TRACE
       ALL
       CONDITION INTERFACE <nom_interfaz>
       CONDITION GLOBAL-BUFFERS
       CONDITION PROTOCOL <nom_protocolo>
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
```

Note

Enabling an event as a trap using the **snmp-trap** command enables it for all trap groups.

Example 1:

```
ELS config>enable TRACE EVENT ICMP.001
ELS config>
```

This example enables a single event (ICMP.001), causing it to be shown on screen.

Example 2:

```
ELS config>enable SYSLOG GROUP   MYGROUP
ELS config>
```

This example enables MYGROUP events so that they are sent as syslog messages.

Example 3:

```
ELS config>enable ALL SUBSYSTEM IP INFO
ELS config>
```

This example enables IP subsytem INFO filter level events to be displayed on screen, sent as SNMP traps and syslog messages.

Important

Do not run this command for long periods of time when the device is transferring packets because a significant amount of time will be lost in communicating with the VISEVEN process. Running it when communicating with the **Teldat Router** via a remote terminal may cause the device to spend most of its time communicating with the remote terminal.

4.4.1.9 ENVIRONMENT-MONITOR

Allows you to monitor the temperature probes, fans and PSUs to detect when they fail. Regardless of whether any errors are detected, this option also allows you to read these parameters and send their values to NSLA filters.

Syntax:

```
ELS config>environment-monitor ?
  report            Report info to NSLA system
  periodic-event    Sends periodic events when the sensors indicate a value that exceeds safe levels.
                    To check the thresholds, use the 'system health' command under monitoring.
```

> **Note**
>
> This command is only available on devices equipped with temperature probes and/or that have fan and/or PSU monitoring capabilities.

Release	Modification
11.00.05	The **report** option was introduced as of version 11.00.05.
11.01.01	The **report** option was introduced as of version 11.01.01
11.00.07	The help text of the **periodic-event** option has been changed.
11.01.05	The help text of the **periodic-event** option has been changed.
11.01.01.60.06	The help text of the **periodic-event** option has been changed.

4.4.1.9.1 ENVIRONMENT-MONITOR REPORT

Allows you to report the read values to a NSLA filter every so often, thus allowing further monitoring of environment parameters.

Syntax:

```
ELS config>environment-monitor report [report-type] [report-id] nsla-filter [filter-number] /
           interval-sec [seconds]
```

Examples:

> **Note**
>
> Device performance may be affected if the interval is too low. This is because updating the reported values requires a large number of internal requests.

```
ELS config>environment-monitor report fan-speed 3 nsla-filter interval-sec 60
```

In this example, the *CASE3* fan speed values are reported to NSLA filter 3 every 60 seconds.

```
ELS config>environment-monitor report psu-voltage 1 nsla-filter 1 interval-sec 30
```

In this example, the PSU 1 voltage values are reported to NSLA filter 1 every 30 seconds.

```
ELS config>environment-monitor report temperature CPU_EXT nsla-filter 2 interval-sec 10
```

In this example, the temperature value from the CPU_EXT thermal probe is reported to NSLA filter 2 every 10 seconds.

Additional information:

The limitations of the report system mean decimal values cannot be reported. Therefore, the **psu-voltage** value is reported in mV (millivolt) units, the **temperature** in ºC, and **fan-speed** in rpm.

Configuration example:

> **Note**
>
> Adding an appropriate NSLA filter configuration is extremely important. There are many possible configurations for different scenarios and requirements. In the end, it's down to the end user to choose the configuration that best suits their needs.

```
ELS config>environment-monitor report psu-voltage nsla-filter 1 interval-sec 30
```

This is the same command that was used in the previous example, but now in the NSLA configuration:

```
   feature nsla
; -- Feature Network Service Level Advisor --
      enable
;
      filter 1 generic-input
      filter 1 significant-samples 100
      filter 1 activation threshold 12000
      filter 1 activation sensibility 100
      filter 1 activation stabilization-time 0
      filter 1 deactivation threshold 11000
      filter 1 deactivation sensibility 50
      filter 1 deactivation stabilization-time 0
      filter 1 initial-status inactive
      filter 1 invert
```

Configuration explanation:

First, we have to set the filter input as **generic-input**. Then, we need to ensure that we select appropriate values in relation to the reported value. In this example, a PSU failure occurs when 50 % or more of the reported values fall below the configured threshold in the configured window of 100 samples. The filter output activates at that point.

When 100 % of the window samples reach 12 V, the filter output deactivates.

Since the **invert** option has to be set to achieve this behavior, **initial-state** is set to inactive to avoid filter output activation before any samples are reported.

4.4.1.9.2 ENVIRONMENT-MONITOR PERIODIC-EVENT

Allows you check the temperature probes, fans and PSUs and generate an event when they have a value which is considered abnormal. This event can be of three types:

- ENV.001 Reports that one of the device's temperature probes is indicating a temperature to above-safe levels.
- ENV.002 Reports that one of the device's fans is indicating a failure.
- ENV.003 Reports that one of the device's PSUs is indicating a voltage abnormality.

These events must be enabled before they can be displayed or sent as traps or syslog messages.

Syntax:

```
ELS config>environment-monitor periodic-event <time_between_events> <trigger_events>
```

```
ELS config>environment-monitor periodic-event ?
   <1m..1d>    Time between events
ELS config>environment-monitor periodic-event 1m ?
   unique    Trigger the events once only
   <cr>      Trigger the events periodically
ELS config>environment-monitor periodic-event 1m unique
```

The checks are performed periodically according to the specified time and the events can be triggered once (when the monitored parameters fail the first time) or periodically (every time they fail).

Example:

```
ELS config>environment-monitor periodic-event 2h
ELS config>
```

In the example, parameter checks are carried out every two hours. In each check, an event is generated if there is a failure.

```
ELS config>environment-monitor periodic-event 1h unique
ELS config>
```

In the example, parameter checks are made every hour and an event is generated only when they fail for the first time.

This command also displays temperature sensor information when running the **configuration** command from the monitoring process.

Example:

```
*monitor
Console Operator

+configuration
[…]
Watchdog timer Enabled
CASE 1 fan speed: 0 rpm (0 %)
                     CPU temperature: 51°C
                     ADSL1 temperature: 50°C
                     ADSL2 temperature: 53°C                […]
```

Note

Through the **system health** command, found under monitoring, you can obtain the normal value range for the temperature probes and PSUs.

Release	Modification
11.00.07	The **unique** option was introduced as of version 11.00.07.
11.01.05	The **unique** option was introduced as of version 11.01.05.
11.01.01.60.06	The **unique** option was introduced as of version 11.01.01.60.06.

4.4.1.10 EV-BUFFER <num. Lines> <line size>

Allows you to save an amount of memory for the event buffer. For this to work, you only need to configure two parameters: the number of lines (each event is stored in a line) and the size of each line saved in memory.

Syntax:

```
ELS config>ev-buffer <num_lines> <line_size>
```

Example:

```
ELS config>ev-buffer ?
  <2..10000>    Number of lines
ELS config>ev-buffer 1000 ?
  <28..200>    Line size
ELS config>ev-buffer 1000 130
Please restart to take effect.
ELS config>
```

4.4.1.11 FILTER

Applies a filter.

Event filtering allows you to apply filters to a given event in order to highlight relevant information and filter out irrelevant information. It also has the added advantage of not storing the discarded events, thus reducing the risk of events being lost due to overflow.

The filters have an associated index that determines the order in which filters are applied, so that lower order filters will be applied before higher order filters. The order must be between 1 and 10. At most, you can define 10 filters simultaneously.

Another parameter associated with a filter is the application condition. If met, the filter is applied. Filters are checked one after the other until none are left or until one of them is applied. The condition applies to the event text, not the event identifier. The condition is given by a text string or by a regular expression to look for, as well as a position. Whereas the text must be between double quotation marks, the position can be given explicitly or you can specify a value of -1 to indicate any position.

The last parameter associated with a filter is the action to apply. This can be to exclude, highlight or accept the event, or to start/stop storing events.

Event filtering can be turned on and off globally using the **enable filter** and **disable filter** commands, respectively.

Syntax:

```
ELS config>filter ?
  <1..10>    Entry
ELS config>filter 1 ?
  default     Create a event filter with action exclude and pos ignore
  text        Text to filter
  position    Position
  action      Action to be applied
ELS config>filter 1 text ?
  <0..40 chars>    Text
ELS config>filter 1 action ?
  include         includes the trace
  exclude         excludes the trace
  red             shows the trace in red
  green           shows the trace in green
  yellow          shows the trace in yellow
  blue            shows the trace in blue
  magenta         shows the trace in magenta
  cyan            shows the trace in cyan
  stop-traces     stops saving traces
  start-traces    starts saving traces
ELS config>
```

Example 1:

A simple example of event filtering is IP debugging on a device accessed via Telnet: if the IP events are enabled, the events that you want will appear, but so too will the Telnet events. This means you will end up with a large number of IP events. To way to get around this it to enable event filtering to exclude all events that carry the Telnet client's IP address (172.24.78.94).

```
ELS config>filter 1 text "172.24.78.94"
```

Example 2:

Shows the GW.019 event (an internal event which occurs when an interface performs a self-test) in red.

```
ELS config>filter 1 text "GW.019"
ELS config>filter 1 action red
ELS config>
```

Example 3 (using regular expressions):

Shows the GW.019 event (an internal event which occurs when an interface performs a self-test) in red, but only for the ethernet0/0 interface.

```
ELS config>filter 1 text "GW.019.*ethernet0/0"
ELS config>filter 1 action red
ELS config>
```

Example 4:

Stops storing events after the first event with IP address 192.168.212.116 is detected.

```
ELS config>filter 1 text "192.168.212.116"
ELS config>filter 1 action stop-traces
ELS config>
```

4.4.1.12 LIST

Lists information about enabled events, created groups, subsystems and settings.

Syntax:

```
ELS config>list ?
  all                   Lists the configuration and all the subsystems
  configuration         Lists the status of the subsystems, groups and events
  ev-buffer             Lists the events buffer parameters
  event                 Lists the filter level and the specified event message
  filter                Lists status of the filtering and the configured
                        filters
  groups                Lists the groups defined by the user and their content
  nvram-log-priority    Lists the minimum priority of logs saved
  subsystem             Lists all the events of a specified subsystem
ELS config>
```

LIST ALL

This command lists: the defined groups and their events; the configuration status of each individual subsystem, group and event; parameters relating to the size of the event buffer for events waiting to be displayed on screen; possible event filters established and whether or not filtering is enabled; and lastly, the minimum log priority for storing logs in non-volatile memory.

Example:

```
ELS config>list all
Group: MYGROUP
       IP.002
       IP.003
       IP.004
Subsystem     :GW
    Trace     :ALL
    Syslog    :ALL
    SNMP-Trap (all groups):ALL
Subsystem     :IP
    Trace     :STANDARD
    Syslog    :none
    SNMP-Trap (all groups):none
Group         Trace     Syslog    SNMP-Trap
MYGROUP       Off       On        On ( group 1 group 3 )
Event         Trace     Syslog    SNMP-Trap
ICMP.001      On        Off       On ( all groups )
Events Buffer Parameters:
Number of lines: 50  Line size: 208
 EVENT FILTER
 Events filtering DISABLE

 { num) string, /pos -> action }
  1) 172.24.78.94 /-1 -> exclude
  2) Rx /1 -> red

Minimum priority of logs saved: Priority 5
ELS config>
```

LIST CONFIGURATION

Lists the status (enabled/disabled) of individual subsystems, groups and events that have been configured and that will be used after the next reboot if they've been stored to memory.

Assuming that the GW subsystem has been enabled to send SNMP traps and syslog messages and for them to be viewed from the VISEVEN process, that the IP subsystem has been enabled to display STANDARD filter level events on screen, that the MYGROUP group has been enabled to accept syslog notifications, and that the user has enabled the ICMP:001 event to be sent as a company-specific trap, then we would get the result below:

Example:

```
ELS config>list configuration
Subsystem     :GW
    Trace     :ALL
    Syslog    :ALL
    SNMP-Trap (all groups):ALL
Subsystem     :IP
    Trace     :STANDARD
    Syslog    :none
    SNMP-Trap (all groups):none

Group         Trace     Syslog    SNMP-Trap
MYGROUP       Off       On        On ( group 1 group 3 )

Event         Trace     Syslog    SNMP-Trap
ICMP.001      Off       Off       On (all groups )
ELS config>
```

LIST EV-BUFFER

Lists parameters related to the size of the storage buffer, which holds the events waiting to be displayed on the screen.

Example:

```
ELS config>list ev-buffer
Events Buffer Parameters:
Number of lines: 1000  Line size: 300
ELS config>
```

LIST EVENT

Lists the specified event's filter level and message.

Example:

```
ELS config>list event ICMP.001
Level: UE-ERROR
Message: bd cks 0x%04x (exp 0x%04x) %I -> %I
ELS config>
```

LIST FILTER

Lists information about event filtering: general filter status and configured filters.

Example:

```
ELS config>list filter
 EVENT FILTER
 Events filtering DISABLE
 { num) string, /pos -> action }
  1) 172.24.78.94 /-1 -> exclude
  2) Rx /1 -> red
ELS config>
```

LIST GROUPS

Lists the names of user-defined groups and their contents.

Example:

```
ELS config>list groups
Group: MYGROUP
       IP.002
       IP.003
       IP.004
ELS config>
```

LIST SUBSYSTEM

Lists all the events of a specified subsystem.

Example:

```
ELS config>list subsystem icmp
Event         Level     Message

ICMP.001      UE-ERROR  bd cks 0x%04x (exp 0x%04x) %I -> %I
ICMP.002      C-INFO    ech %I -> %I
ICMP.003      U-INFO    ech rp %I -> %I
ICMP.004      CI-ERROR  unhnd typ %d %d  %I -> %I
ICMP.005      U-TRACE   unhnd brd typ %d %d  %I -> %I
ICMP.006      UE-ERROR  bd typ %d %d  %I -> %I
ICMP.007      C-TRACE   addr msk %I -> %I
ICMP.008      C-TRACE   addr msk rep %I -> %I
ICMP.009      UI-ERROR  no pkt or mem
ICMP.010      UE-ERROR  amb addr msk %I -> %I
ICMP.011      UI-ERROR  err %d sndng pkt to ifc %s
ICMP.012      C-INFO    rdr %I -> %I to %I
ICMP.013      U-INFO    bd prm off %d %I -> %I
ICMP.014      U-TRACE   snd %d %d pkt %I -> %I
ICMP.015      UE-ERROR  shrt ICMP hdr %d src %I
ICMP.016      U-TRACE   %I rdr dest %I to %I
ICMP.017      UE-ERROR  Bad rdr from %I, rsn: %S
ICMP.018      U-TRACE   Router advertisement received from %I
ICMP.019      UE-ERROR  Bad router adv from %I, rsn: %S
ICMP.020      U-TRACE   rcvd typ %d %d  %I -> %I
ICMP.021      P-TRACE   redirect message filtered at interface %s
ICMP.022      P-TRACE   unreachable message filtered at interface %s
ELS config>
```

If no subsystem name is entered, the system lists the name, number of events and description for all the subsystems.

Example:

```
ELS config>list subsystem
Name      Events    Description

ADSL      8         ADSL
AFS       6         Advanced Filtering Subsystem
AINST     23        AutoInstall
ARP       10        Address Resolution Protocol
ASDP      11        Asynchronous Serial Device Proxy
ARLY      41        Alarm Relay
ASYN      5         Asynchronous Serial Line
AT        20        AT Commands Interface
ATM       15        Asynchronous Transfer Mode
BAN       29        Boundary Access Node
BGP       27        Border Gateway Protocol
BR        48        Bridge/Routing
BRS       9         Bandwidth  Reservation
BSPF      10        Bridge Spoofing
CIF       34        Encryption
CNSL      4         Console
DEP       30        DEP Forwarder
DHCP      14        DHCP
DHCPC     23        DHCP Client
DLS       459       Data Link Switching
DNAT      12        Dynamic NAT
DNS       30        Domain Name System
EAP       6         EAP
ETH       54        Ethernet
FLT       7         Filter Library
FR        53        Frame Relay
FRBK      8         Frame Relay BACKUP
FTP       4         File Transfer Protocol
G703      25        G703 Digital Interface
GW        64        Router kernel
H323      19        H323
HDLC      11        HDLC Interface
HDSL      57        Symetric High Bitrate Digital Subscriber Line
HSSI      5         High Speed Serial Interface
HOTSPOT   39        HotSpot
HTTP      25        HyperText Transfer Protocol
ICMP      22        Internet Control Message Protocol
IGMP      26        Internet Group Management Protocol
IKE       51        Internet Key Exchange
IP        87        Internet Protocol
IP6       200       IPv6
IPHC      46        IP Header Compression
IPSEC     33        Ip Security
IPX       105       Internetwork Packet Exchange Protocol
ISDN      40        Integrated Services Digital Net
L2TP      56        Layer 2 Tunneling Protocol
LAPD      11        ISDN Layer 2
LDAP      16        Lightweight Directory Access Protocol
LLC       33        Logical Link Control
MCF       9         MAC Filtering
NAPT      30        Network Address Port Translation
NBS       50        NetBIOS Support Subsystem
NHRP      58        Next Hop Resolution Protocol
NOE       17        NOE
NSLA      8         Network Service Level Advisor
NSM       82        Network Service Monitor
NTP       25        Network Time Protocol
P3OE      23        PPP over Ethernet
PHYS      4         ISDN BRI Layer 1
PGMO      5         POS Gateway Monitor
POLR      16        Policy routing
```

```
PPP       100       Point to Point Protocol
R2        9         R2
RAD       46        Remote Authentication Dial In User Service
RIP       30        IP Routing Information Protocol
RSTP      9         Rapid Spanning Tree Protocol
SCADA     28        SCADA Network
SCDFW     20        SCADA Forwarder
SCEP      17        Simple Certificate Enrollment Protocol
SDLC      95        IBM SDLC
SIP       16        SIP
SL        36        Serial Line
SMGT      9         System Management
SNAT      5         Static NAT
SNMP      26        Simple Network Management Protocol
SPF       61        Open SPF-Based Routing Protocol
SRT       89        Source Routing Transparent Bridge
STP       42        Spanning Tree Protocol
STUN      3         STUN
SYNC      2         Synchronous Serial Line
TCP       55        Transmission Control Protocol
TIDP      18        T. IP Discovery Protocol
TKR       46        Token Ring
TLNT      8         Telnet
TLPHY     23        TLPHY
TNIP      39        IP Tunnel
TTTP      18        T. Transaction Transfer Protocol
TVRP      26        T. Virtual Router Protocol
UDAFO     41        UDAFO Forwarder
UDP       4         User Datagram Protocol
VOIP      14        Voice over IP
VRRP      8         Virtual Router Redundancy Protocol
WLAN      4         Wireless LAN
X252      23        X.25 Layer 2
X253      26        X.25 Layer 3
XN        21        Core Xerox Network System

ELS config>
```

Command history:

Release	Modification
11.00.05	The SMGT event was introduced as of version 11.00.05.
11.01.00	The SMGT event was introduced as of version 11.01.00.

LIST NVRAM-LOG-PRIORITY

Lists the minimum priority of the logs that are stored in the non-volatile memory.

Example:

```
ELS config>LIST NVRAM-LOG-PRIORITY
Minimum priority of logs saved: Priority 5
ELS config>
```

4.4.1.13 NO

Allows you to delete a given filter or reset the default storage buffer size for events waiting to be displayed on screen, or the default minimum logging level for logs that will be stored in non-volatile memory.

It also allows you to delete and reset the configuration on the enabled or disabled events at any time.

Syntax:

```
ELS config>no ?
  environment-monitor   Enables environment monitor
  disable               Disables events
  enable                Enables events
  ev-buffer             Sets default events buffer size
  filter                Eliminates a given filter
  number-events-log     Number of events to be logged in case of fatal error
  nvram-log-priority    Sets the default priority of logs saved
  print                 Configure els show options
  stop-traces           Stops saving traces
  trace-level           Configures subsystem trace level
ELS config>
```

NO ENVIRONMENT-MONITOR

Release	Modification
11.00.05	The **report** option was introduced as of version 11.00.05.
11.01.01	The **report** option was introduced as of version 11.01.01
11.00.07	From now on, this functionality can not be disabled in general, now each option of this command has its own 'no'.
11.01.05	From now on, this functionality can not be disabled in general, now each option of this command has its own 'no'.
11.01.01.60.06	From now on, this functionality can not be disabled in general, now each option of this command has its own 'no'.

Allows you to disable monitoring temperature probes, fans and PSUs to detect when they fail, and to disable reading these parameters (to avoid sending their values to NSLA filters).

Syntax:

```
ELS config>no environment-monitor ?
  report           Report info to NSLA system
  periodic-event   Sends periodic events when the sensors indicate a value that exceeds safe levels.
                   To check the thresholds, use the 'system health' command under monitoring.
```

Example 1:

```
ELS config>no environment-monitor periodic-event
ELS config>
```

This example disables all temperature probe, fan and PSU checks that generate events if anomalous behavior is detected.

Example 2:

```
ELS config>no environment-monitor report temperature POWER
ELS config>
```

This disables reading the POWER temperature probe (meaning its value will not be sent to an NSLA filter).

NO DISABLE

Selects and resets the default parameters for (disabled) events displayed on the screen, or sent as traps or syslog messages. Groups and subsystems can be restored.

Example:

```
ELS config>no disable TRACE GROUPS miGrupo
ELS config>
```

This example resets the miGrupo group's events to their default values so they are displayed on the screen, and clears the corresponding entry from the configuration.

NO ENABLE

Selects and resets the default parameters for (enabled) events displayed on the screen, or sent out as traps or via syslog messages. Groups and subsystems can be restored.

Example:

```
ELS config>no enable SNMP-TRAP SUBSYSTEM ARP ALL
ELS config>
```

This example resets ARP subsystem events with the filter level ALL to their default values so they are sent out as SNMP traps, and clears the corresponding entry from the configuration.

Example:

```
ELS config>no enable SYSLOG EVENT IP.007
ELS config>
```

This example resets the single event IP.007 to its default value so that it is sent as a syslog message, and clears the corresponding entry from the configuration.

NO EV-BUFFER

This resets the storage buffer parameters for events waiting to be displayed as console traces (VISEVEN process) to their default values. These parameters allow you to choose the amount of reserved memory to use. These default values are 50 lines or messages and 208 bytes per line.

Example:

```
ELS config>no ev-buffer
ELS config>
```

NO FILTER

Deletes a previously configured filter.

Example:

```
ELS config>no filter 2
ELS config>
```

NO PRINT

Suppresses the printing of additional information with each previously configured event.

Example:

```
ELS config>no print extra-info
ELS config>
```

NO STOP TRACES

Eliminates the order to stop storing events.

Example:

```
ELS config>no stop-traces
ELS config>
```

NO NUMBER-EVENTS-LOG

Sets the default value for the number of events that are stored in non-volatile memory when a device RESET occurs.

Example:

```
ELS config>no number-events-log
ELS config>
```

NO NVRAM-LOG-PRIORITY

Sets the default minimum logging level for logs that are stored in non-volatile memory (priority 5).

Example:

```
ELS config>no nvram-log-priority
ELS config>
```

4.4.1.14 NUMBER-EVENTS-LOG

Configures the number of events that are logged in non-volatile memory when a device RESET occurs. The default value is 3 and the configurable values range from 3 to 10,000. Knowing the last events generated before an error occurs is very useful when it comes to detecting the cause of the error. Setting a higher value than the default value is therefore advisable, as it will provide you with more information with which to diagnose the problem.

Example:

```
ELS config>number-events-log ?
  <3..10000>    value in the specified range
```

```
ELS config>number-events-log 100
ELS config>
```

4.4.1.15 NVRAM-LOG-PRIORITY

Sets the minimum priority logs must have to be stored. Each log is given a priority of 1 to 5 (1 being the highest priority and 5 the lowest). Given the limited capacity of non-volatile memory (logs are stored in a circular queue), you might prefer to store only the highest priority logs and not the lowest priority ones.

Logs relating to device startups have priorities between 1 and 3. All other logs (device access, configuration changes, etc.) are assigned priorities between 4 and 5.

Example:

```
ELS config>nvram-log-priority ?
  <1..5>    Minimum priority of logs saved
ELS config>nvram-log-priority 5
ELS config>
```

4.4.1.16 OPTIONS

Allows you to modify the behavior of the Event Logging System (ELS).

Syntax:

```
ELS config>options ?
  circular-buffer        Enable circular buffer
  no                     Negate options
  time-as-incremental    Show time as incremental in milliseconds
  time-as-ticks          Show time as ticks
  time-stamp             Include time in events
  usb-save               Save events on external USB
ELS config>
```

CIRCULAR-BUFFER

Sets the default behavior of the Event Logging System for logging events to the circular buffer. New events are always added to the circular buffer and, if the buffer is already full, older events are discarded.

NO

Allows you to disable certain options that are enabled by default. These include:

4.4.1.16.1 CIRCULAR BUFFER

The Event Logging System stops entering events in the circular display buffer when it is full. When events are not displayed and the buffer is full, CPU time is not consumed, so the system can increase its performance. The side effect is that the most recent events are lost, not the oldest.

4.4.1.16.2 TIME-AS-INCREMENTAL

Removes milliseconds from the elapsed time since the the arrival of the last event.

4.4.1.16.3 TIME-AS-TICKS

Turns off tick marks.

4.4.1.16.4 TIME STAMP

The date and time are not displayed, so the real-time clock is not queried, the probability of losing events decreases, and performance improves.

Example:

```
ELS config>options not-time
 ARP.002 Pkt in 1 1 800 ethernet0/0
 ARP.008 rcv IP RQST 172.24.0.203->172.24.0.206 ifc ethernet0/0
 ARP.002 Pkt in 1 1 800 ethernet0/0
 ARP.008 rcv IP RQST 172.24.76.2->172.24.0.25 ifc ethernet0/0
 ARP.002 Pkt in 1 1 800 ethernet0/0
 ARP.008 rcv IP RQST 172.24.79.3->172.24.78.99 ifc ethernet0/0
```

4.4.1.16.5 USB-SAVE

Disables dumping events to an external USB mass storage device. Please see the **usb-save** command options.

TIME-AS-INCREMENTAL

Events are timestamped with the elapsed time (in milliseconds) between the arrival of the last event and the arrival of the next event.

Example:

```
ELS config>options time-as-incremental
       200.892508ms  IP.007 192.168.212.4 -> 192.168.212.116
         0.007363ms  POLR.004 dis int ethernet0/0
         0.007878ms  IP.036 rcv pkt prt 1 frm 192.168.212.4
         0.007090ms  ICMP.002 ech 192.168.212.4 -> 192.168.212.116
         0.008454ms  IP.061 add lcl pkt to ip op q
         0.009939ms  IP.007 192.168.212.116 -> 192.168.212.4
```

TIME-AS-TICKS

Events are timestamped with ticks instead of with the date and time, which provides greater accuracy.

Example:

```
ELS config>options time-as-ticks
   4065675.488814ms  IP.036 rcv pkt prt 1 frm 192.168.212.4
   4065675.495965ms  ICMP.002 ech 192.168.212.4 -> 192.168.212.116
   4065675.504844ms  IP.061 add lcl pkt to ip op q
   4065675.514450ms  IP.007 192.168.212.116 -> 192.168.212.4
```

TIME-STAMP

Sets the default behavior of the Event Logging System for displaying time stamps.

USB-SAVE

Allows you to store the device's active events in a mass storage device connected to an external USB. Events are stored in a file called **event.log** in the root directory of the first detected partition. The partition must be formatted with a FAT or VFAT file system. If the file exists, the new events are added to the end by inserting a session start mark. Existing information is never deleted.

Example:

```
################################################################################
#                          New log session started                             #
################################################################################
01/01/00 00:00:16 GW.001 System restarted -- H1+ WAN IPSec router cold start
01/01/00 00:00:16 GW.002 Portable CGW H1+ WAN IPSec Rel 10.08.29-Alfa strtd
01/01/00 00:00:17 GW.005 Bffrs: 1471 avail 1441 idle   fair 114 low 294
```

If this command is configured in the router's startup configuration, the USB device must be connected before starting the router. If it does not detect a connected device, the dump operation will not take place even if you connect a device later.

The command can be enabled/disabled (option **no save-usb**) dynamically. The USB device must be connected to the external connector before enabling the command dynamically. If a mass storage device is not connected to the USB connector, the dynamic command fails. If the event dump function is enabled on the external USB device, you can disable the dump by running the **no save-usb** options in the dynamic configuration. You should always dynamically disable event dumps to USB before removing the external storage device. If it is extracted without first disabling the dump, the most recent events are lost and you won’t be able to restart the dump dynamically. You will also have to restart the device.

By default, event dumps to an external USB device are disabled.

Example:

```
ELS config>options usb-save
```

Note

This command is only available on devices with a USB interface.

4.4.1.17 PPP

Allows you to access the PPP event menu. For more information, please see the following manual: *Teldat Dm710-I PPP Interface.*

Example:

```
ELS config>ppp
--  PPP Events Configuration  --
PPP Events config>
```

4.4.1.18 PRINT

Allows you to print additional information with each event. This information is printed on a separate line from the main event and does not include the date and time.

PRINT EXTRA-INFO

Prints the vrf, and the input and output interfaces.

Example:

```
04/15/08 13:13:17  AFS.001 IP IN src 172.24.100.130 dst 172.24.100.129 prt TCP in ifc
ethernet0/0 -> ACCEPTED
                   AFS.001 [vrf: <main> inifc: ethernet0/0 outifc: ppp1]
```

PRINT IP-HEADER

Prints the IP header. Not all events will contain an IP header. In events where it is not available, no additional information is displayed.

Example:

```
04/15/08 13:13:17  AFS.001 IP IN src 172.24.100.130 dst 172.24.100.129 prt TCP in ifc
ethernet0/0 -> ACCEPTED
```

PRINT MISMATCH

Prints additional information about an event only when there is a mismatch with the last printed one. This is useful to avoid redundant information on related events and make them more readable.

Example:

```
ELS config>print mismatch
```

Command History:

Release	Modification
11.01.09	The "*print mismatch*" command was introduced as of version 11.01.09

4.4.1.19 STOP-TRACES

Stops the event log. Although events are not stored, they are analyzed to determine whether they match any of the configured filters.

This can be useful if we want the Event Logging System (ELS) to start tracing when a certain event occurs. You would boot the router with the ELS disabled and you would configure a filter to detect the desired event and associate it with the **start tracing** action.

4.4.1.20 TRACE-LEVEL

Allows you to set the trace level that is displayed for a subsystem. The minimum level (fewest traces) is *error* and the highest level (most traces) is *excessive*.

Event level is not available for all event subsystems, only for some.

The available trace levels are:

- *error*: messages for serious errors that can cause the associated functionality to stop working.
- *warning*: messages for unexpected errors that occur when the associated functionality is running.
- *info*: traces about the normal operation of the associated functionality.
- *debug*: debugging traces.
- *msg-dump*: shows traces related to the messages exchanged during the operation of the associated functionality.
- *excessive*: shows all available traces.

Each level includes the previous ones. If, for example, the debug level traces are enabled, the *info, warning* and *error* level traces are also included.

Syntax:

```
ELS config>trace-level <subsystem> <error-level>
```

Example:

```
ELS config>trace-level wlan debug
```

4.4.1.21 VRF

Allows you to access the Event Filter menu by VRF.

Example:

```
ELS config>vrf
-- VRF Events Configuration --
VRF Filter Events Config>?
```

Within this menu, we can configure the VRFs for which we want to display events.

Syntax:

```
ELS config>vrf

-- VRF Events Configuration --
VRF Filter Events Config>vrf ?
  <1..32 chars>     VRF name
```

Example:

```
VRF Filter Events Config>vrf vrf-1
VRF Filter Events Config>vrf vrf-2
```

4.4.1.22 EXIT

Exits the Event Logging System configuration and returns to the *Config>* prompt.

Syntax:

```
ELS config>exit
```

Example:

```
ELS config>exit
Config>
```

4.4.2 Monitoring process commands

These commands are executed in the monitoring process at the *ELS+* prompt.

Changes made in this process run automatically and are lost when the device is restarted. These commands allow you to enable events at runtime.

Command	**Function**
? (HELP)	Lists all the commands for monitoring the Event Logging System.
CLEAR-ACTIVES	Disables all enabled events at a given time.
CONDITION-DEBUG	Displays useful information for debugging the conditional events functionality.
DISABLE	Allows you to disable event messages so that they are not shown on the screen,

sent as syslog messages or as company-specific traps.

ENABLE Allows you to enable event messages so that they are shown on the screen, sent as syslog messages or company-specific traps.

EVENT-STORE Stores event messages in flash memory when CIT crashes.

FILTER Specifies filtering criteria so only currently relevant events are shown at that time.

HIDE Prevents events from being displayed in the active process. From this moment, events can only be viewed from the VISEVEN process.

This command is available from any process, even though it does not appear when requesting a list of all available commands. It must be written in full.

LIST Lists displays event information and messages.

NO Disables an option.

NVRLOG Allows you to view the logs stored in non-volatile memory, and to initialize the log.

RESTORE-CONFIGURATION Allows you to activate the configuration of the currently existing Event System (previously entered in the corresponding menu of the configuration process) without having to save and restart the device.

SHOW-STORED-LOG Displays stored log information, generated when CIT crashes and the event-store option is enabled.

VIEW Allows events to be displayed in the active process. From this moment, they can no longer be viewed from the VISEVEN process.

This command is available from any process, even though it does not appear when requesting a list of all available commands. It must be written in full.

EXIT Exits event monitoring.

4.4.2.1 ? (HELP)

Lists available commands from the current prompt. You can also enter a question mark (?) after a specific command to list its options.

Syntax:

```
ELS+?
```

Example 1:

```
ELS+?
  clear-actives            Disable all enabled events at a given time
  condition-debug          Monitoring options of condition events debugging
  disable                  Disable event messages
  enable                   Enable event messages
  filter                   Configure the rules permitting events filtering
  hide                     Prevent the events from being displayed
  list                     List information on established events and messages
  nvrlog                   View/Initialize the logs stored in the non-volatile
                           memory
  restore-configuration    Activate the current Events Logging System
                           configuration
  view                     Display the events in the active process
  exit
ELS+
```

Example 2:

```
ELS+LIST ?
  active       List the enabled events of the specified subsystem
  condition    List the interfaces where packet marking has been enabled
  event        List event information
  groups       List group information
  subsystem    List subsystem information
ELS+
```

4.4.2.2 CLEAR-ACTIVES

Allows you to disable all enabled events at a given time.

Syntax:

```
ELS+clear-actives
```

Example:

```
ELS+clear-activesdetain
Do you want to disable all active events?(Y/N)(N): y
All events disabled
ELS+
```

4.4.2.3 CONDITION-DEBUG

Provides useful information if you want to debug the conditional events functionality seen in *Event Logging System user interface* on page 158. To exclude events generated by packets that have not been tagged, a semaphore object is used to tell the system when a packet is tagged. The semaphore is blocked when a packet is tagged because one of the enabled conditions is met, and its unlocked when the packet is no longer tagged.

Syntax:

```
ELS+condition-debug ?
  semaphore-clear    Clear the condition semaphore
  semaphore-dump     Dump information on the condition semaphore
```

The **semaphore-clear** option is used to manually unlock the semaphore, thus allowing another packet to be tagged when one of the enabled conditions is met.

Using the **semaphore-dump** option will give you the following information on the state of the semaphore:

- *Semaphore state*: shows 1 if the semaphore is locked and 0 otherwise.
- *Buffer address*: memory address where the tagged packet that blocked the semaphore is stored.
- *Trigger condition*: condition that caused the packet to be tagged.
- *Calls sequence*: sequence of calls in the program code through which the semaphore was blocked.

Example:

```
ELS+condition-debug semaphore-dump
==================================
..: Condition events semaphore :..
==================================
Semaphore state: 1
Buffer address: 0x01d1cd8c
Trigger condition: interface ethernet0/0
Calls sequence: 00640490<-0066AD14<-00B62830<-00D62E20<-00D63044<-

ELS+
```

4.4.2.4 DISABLE

Selects and disables events so that their messages are not displayed on the screen when using the VISEVEN process, or sent out as SNMP traps or via syslog messages. Groups, subsystems, and all traces (provided they were previously enabled with the **enable trace all** command) can be disabled. You can also use the command to disable the use of event filters.

If you have selected an input interface to display all the traces associated with the processing and path the packets received on that interface follow, the **disable trace condition interface <interface name>** command allows you to disable packet tagging on that interface.

The same procedure is used to disable the global buffer and protocol conditions with the **disable trace condition global-buffers** and **disable trace condition protocol <protocol name>** commands, respectively.

Syntax:

```
ELS+disable
ALL
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SYSLOG
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
FILTER
```

```
TRACE
       ALL
       CONDITION INTERFACE <interface name>
       CONDITION GLOBAL-BUFFERS
       CONDITION PROTOCOL <nom_protocolo>
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
```

Example 1:

```
ELS+disable trace event icmp.001
ELS+
```

This example disables the individual event ICMP.001 so it is not displayed on the screen.

Example 2:

```
ELS+disable syslog group mygroup
ELS+
```

This example disables the MYGROUP group so that events belonging to the group are not sent as syslog messages.

Example 3:

```
ELS+disable all subsystem ip info
ELS+
```

This example disables INFO filter level events from the IP subsystem so that they are not displayed or sent as SNMP traps or as syslog messages.

4.4.2.5 ENABLE

Selects and enables events so their messages are displayed on screen, sent out as traps or transmitted via syslog messages. Groups and subsystems can be enabled. It also allows you to enable the use of event filters.

Under the present chapter's *Event Logging System user interface* on page 158section, we explained how to activate the conditional events functionality. In brief: you use the **enable trace condition interface <interface name>** command to enable packet tagging for a given interface so that the events associated with the processing of packets received on that interface are shown as traces. We also indicated that you can enable all available events to be displayed as traces using the **enable trace all** command. We do not recommend using this command unless the conditional event functionality is used to restrict the number of traces shown to only one packet. Otherwise, the device shows all of them and consequently impacts performance.

In the same section we indicated that you can specify a trigger condition to enable tracing when the free global buffers are used (usually locally sourced packets) or when a packet is processed by a particular protocol, using the **enable trace condition global-buffers** and **enable trace condition protocol <protocol name>** commands, respectively.

Syntax:

```
ELS+enable
ALL
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SYSLOG
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
FILTER
TRACE
       ALL
       CONDITION INTERFACE <interface name>
       CONDITION GLOBAL-BUFFERS
       CONDITION PROTOCOL <nom_protocolo>
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP-GROUP1
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP-GROUP2
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP-GROUP3
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
SNMP-TRAP-GROUP4
       EVENT <subsystem.event_num>
       GROUPS <nom_group>
       SUBSYSTEM <subsystem> <filtered_layer>
```

Note

Enabling an event as a trap using the **snmp-trap** command enables it for all trap groups.

Example 1:

```
ELS+enable trace event icmp.001
ELS+
```

This example enables the individual event ICMP.001 to be displayed on screen.

Example 2:

```
ELS+enable syslog group mygroup
ELS+
```

This example enables the MYGROUP group so that events belonging to the group are sent via syslog messages.

Example 3:

```
ELS+enable all subsystem ip info
ELS+
```

This example enables INFO filter level events from the IP subsystem so that they are displayed on screen, sent out as SNMP traps, and via syslog messages.

Important

Do not run this command for long periods of time when the device is transferring packets, because a significant amount of time will be lost in communicating with the VISEVEN process. Running it when communicating with the **Teldat Router** via a remote terminal may cause the device to spend most of its time communicating with the remote terminal.

4.4.2.6 EVENT-STORE

Stores the most recent event in the event buffer to flash if the OS crashes. It is used for debugging purposes. An in-depth knowledge of Teldat Router software is required to use this command. Do not use this command unless instructed to do so by Teldat personnel.

Syntax:

```
ELS+event-store ?
  <cr>
ELS+
```

When using this command, a message on possible operational risks will appear as warning. If the command is confirmed, the event store is enabled and the event buffer will be saved to flash if the OS crashes.

```
ELS+event-store
WARNING: This command may cause future performance or behaviour issues
Do you want to enable event-store?
Enable event-store(Yes/No)?yes
Event store is enabled. EVENTLOG.EV will be generated when CIT crashes
ELS+
```

If a previous log was generated, the command issues a warning and asks for confirmation.

```
ELS+event-store
WARNING: This command may cause future performance or behaviour issues
Do you want to enable event-store?
Enable event-store(Yes/No)?yes
Event store is enabled. EVENTLOG.EV will be generated when CIT crashes
There is a previous event log saved
Are you sure to enable it?(This action may delete the previous one)
Enable event-store(Yes/No)? yes
ELS+
```

Command History:

Release	Modification
11.00.05	This command was introduced as of version 11.00.05
11.01.01	This command was introduced as of version 11.01.01.

4.4.2.7 FILTER

Allows you to add, change or remove individual filters. Use the **enable filter** command to enable a filter.

For more information, please see the **filter** command in the event configuration menu.

Syntax:

```
ELS+filter
  add     Add/Change filters individually
  del     Delete filters individually
  list    Display the status of events filtering
ELS+
```

FILTER ADD

Adds an entry to the filter table. If the entry already exists, it is replaced.

Syntax:

```
ELS+filter add <entry_number> <text> <position> ACTION <action>
```

Example:

```
ELS+filter add 2 "rx" 1 action red
ELS+
```

FILTER DEL

Deletes an entry from the filter table.

Syntax:

```
ELS+filter del <entry_number>
```

Example:

```
ELS+filter del 2
```

FILTER LIST

Lists the filter table.

Syntax:

```
ELS+filter list
```

Example:

```
ELS+filter list
        EVENT FILTER
State: disabled
 1) 172.24.78.94 /-1 -> Excl
 2) rx /1 -> Red
 3) --- - ---
 4) --- - ---
 5) --- - ---
 6) --- - ---
 7) --- - ---
 8) --- - ---
 9) --- - ---
10) --- - ---

ELS+
```

4.4.2.8 HIDE

Prevents events from being displayed in the active process. From now on, events can only be viewed when using the VISEVEN process.

This command is available from any process, even though it does not appear when requesting all available commands. It must be written in full.

Syntax:

```
ELS+hide
```

Example:

```
ELS+hide
ELS+
```

4.4.2.9 LIST

Lists information about enabled events, created groups, and subsystems. It also shows the interfaces being used with the conditional events functionality.

Syntax:

```
ELS+list ?
  active       List the enabled events of the specified subsystem
  condition    List the interfaces where packet marking has been enabled
  event        List event information
  groups       List group information
  subsystem    List subsystem information
ELS+
```

LIST ACTIVE

Example:

```
ELS+list active arp
Actives      Count   Trace   Syslog  Snmp-Trap
ARP.001      0       on      off     on ( group 1 )
ELS+
```

Lists the enabled events in the ARP subsystem, the number of times each event has occurred, and the enable vector for each event.

Note

Events with the **always** filter level are always enabled to be displayed on the console screen (e.g., GW.001). Similarly, there are events enabled as traps that cannot be disabled because they are responsible for generating the generic SNMP traps (for example, GW.021® link up).

LIST CONDITION

Example:

```
ELS+list condition
Established conditions:
Traces for packets received on ethernet0/0
ELS+
```

Lists the interfaces in which packet tagging has been enabled to track packets and show the traces associated with their processing. For more information, see section 3 (Event Logging System User Interface) or the explanation of the **enable/disable trace condition interface <interface name>** and **enable/disable trace all** commands.

LIST EVENT

Example:

```
ELS+list event icmp.001
Level: UE-ERROR
Message: bd cks 0x%04x (exp 0x%04x) %I -> %I

Count:  0 Status: enable as  (Trace) (Syslog message) (SNMP Trap)
ELS+
```

Lists ICMP.001 event information.

If we assume that the event is currently enabled to be displayed on the screen, sent as a syslog message and as a company-specific trap, then we would get the information shown in the example:

LIST GROUP

Example:

```
ELS+list group
Group: MYGROUP
       Event        Trace   Syslog  Snmp-Trap
       IP.002       on      off     on ( all groups )
       IP.003       on      off     on ( group 2 group 4 )
       IP.004       on      off     off
Globally enable as:  (Trace)
ELS+
```

Displays the name of the group and its events, the current enable status of each event, and the current global enable status for the group.

If all of the events in the group are enabled to be displayed on the screen, and some are enabled to be sent out as a trap and a syslog message, then we would get the information shown in the example:

LIST SUBSYSTEM

Example 1:

```
ELS+list subsystem icmp
Event         Level     Message

ICMP.001      UE-ERROR  bd cks 0x%04x (exp 0x%04x) %I -> %I
ICMP.002      C-INFO    ech %I -> %I
ICMP.003      U-INFO    ech rp %I -> %I
ICMP.004      CI-ERROR  unhnd typ %d %d  %I -> %I
ICMP.005      U-TRACE   unhnd brd typ %d %d  %I -> %I
ICMP.006      UE-ERROR  bd typ %d %d  %I -> %I
ICMP.007      C-TRACE   addr msk %I -> %I
ICMP.008      C-TRACE   addr msk rep %I -> %I
ICMP.009      UI-ERROR  no pkt or mem
ICMP.010      UE-ERROR  amb addr msk %I -> %I
ICMP.011      UI-ERROR  err %d sndng pkt to ifc %s
ICMP.012      C-INFO    rdr %I -> %I to %I
ICMP.013      U-INFO    bd prm off %d %I -> %I
ICMP.014      U-TRACE   snd %d %d pkt %I -> %I
ICMP.015      UE-ERROR  shrt ICMP hdr %d src %I
ICMP.016      U-TRACE   %I rdr dest %I to %I
ICMP.017      UE-ERROR  Bad rdr from %I, rsn: %S
ICMP.018      U-TRACE   Router advertisement received from %I
ICMP.019      UE-ERROR  Bad router adv from %I, rsn: %S
ICMP.020      U-TRACE   rcvd typ %d %d  %I -> %I
ICMP.021      P-TRACE   redirect message filtered at interface %s
ICMP.022      P-TRACE   unreachable message filtered at interface %s
ELS+
```

Example 2:

```
ELS>list subsystem
Name      Events    Description

AAA       49        Authentication, Authorization, Accounting
ACT       1         Alsa Custom Trap
ACL       3         Access List
ADSL      8         ADSL
AFS       59        Advanced Filtering Subsystem
ARP       18        Address Resolution Protocol
ASDP      25        Asynchronous Serial Device Proxy
ASYN      5         Asynchronous Serial Line
AT        21        AT Commands Interface
ATM       15        Asynchronous Transfer Mode
BAN       29        Boundary Access Node
BFD       53        Bidirectional Forwarding Detection
BGP       92        Border Gateway Protocol
BR        48        Bridge/Routing
BRS       19        Bandwidth  Reservation
BSPF      10        Bridge Spoofing
CDP       6         Cisco Discovery Protocol
CELL      24        Cellular AT Commands Interface
CFLOW     10        CFLOW
CFM       10        Connectivity Fault Management
CIF       34        Encryption
CNSL      4         Console
DHCP      16        DHCP
DHCPC     23        DHCP Client
DH6C      12        IPv6 DHCP Client
DH6S      9         IPv6 DHCP Server
DLS       459       Data Link Switching
DNAT      13        Dynamic NAT
DNS       30        Domain Name System
DNSU      27        DNS Updater Feature
DOT1X     27        IEEE 802.1X Authentication Protocol
EAP       8         EAP
EOAM      13        Ethernet OAM
ETH       55        Ethernet
FLT       7         Filter Library
FR        53        Frame Relay
FRBK      8         Frame Relay BACKUP
FTP       4         File Transfer Protocol
G703      31        G703 Digital Interface
GW        72        Router kernel
GW104     20        Gateway IEC104
H323      23        H323
HDLC      15        HDLC Interface
HDSL      6         HDSL
HOTSPOT   39        Hotspot
HTTP      25        HyperText Transfer Protocol
I101      12        IEC101 Gateway Network
ICMP      22        Internet Control Message Protocol
ICMP6     14        Internet Control Message Protocol version 6
IGMP      29        Internet Group Management Protocol
IKE       76        Internet Key Exchange
IP        90        Internet Protocol
IP6       92        IPv6
IPHC      46        IP Header Compression
IPSECFT   51        IPSec fault tolerant
IPSEC     35        Ip Security
ISDN      41        Integrated Services Digital Net
L2TP      56        Layer 2 Tunneling Protocol
```

```
LAPD      11        ISDN Layer 2
LDAP      16        Lightweight Directory Access Protocol
LLC       33        Logical Link Control
LLDP      3         Link Layer Discovery Protocol
MCF       9         MAC Filtering
MGCP      23        Media Gateway Control Protocol
MLD6      17        Multicast Listener Discovery for IPv6
MROUTE    6         Multicast Routing
MSDP      42        Multicast Source Discovery Protocol
MTC       18        MTC messages
NAPT      30        Network Address Port Translation
NBS       50        NetBIOS Support Subsystem
ND        117       Neighbour Discovery IPv6
NEIG      13        Neighbour IPv6
NHRP      58        Next Hop Resolution Protocol
NIC       23        NIC Interface
NOE       24        NOE
NSLA      8         Network Service Level Advisor
NSM       83        Network Service Monitor
NTP       42        Network Time Protocol
P3OE      25        PPP over Ethernet
PHYS      4         ISDN BRI Layer 1
PIM       89        Protocol Independent Multicast
PGMO      5         POS Gateway Monitor
POLR      20        Policy routing
PPP       100       Point to Point Protocol
QMI       23        QUALCOMM MSM Interface
R2        9         R2
RAD       59        Remote Authentication Dial In User Service
RIP       36        IP Routing Information Protocol
RIP6      16        RIPng protocol
RSTP      10        Rapid Spanning Tree Protocol
SCADA     29        SCADA Network
SCCP      8         SCCP
SCDFW     20        SCADA Forwarder
SCEP      17        Simple Certificate Enrollment Protocol
SDLC      95        IBM SDLC
SIP       23        SIP
SL        36        Serial Line
SMGT      9         System Management
SNAT      5         Static NAT
SNMP      29        Simple Network Management Protocol
SPF       76        Open SPF-Based Routing Protocol
SPF6      48        Open SPF-Based Routing Protocol version 3
SPI       16        SPI feature
SRT       91        Source Routing Transparent Bridge
SRVP      22        Service Policy
SSH       79        Secure Shell
SSL       1         Secure Socket Layer Subsystem
STP       48        Spanning Tree Protocol
STUN      3         STUN
SYNC      5         Synchronous Serial Line
TCP       55        Transmission Control Protocol
TFTP      6         Trivial File Transfer Protocol
TIDP      18        T. IP Discovery Protocol
TLNT      10        Telnet
TLPHY     33        TLPHY
TNIP      60        IP Tunnel
TVRP      29        T. Virtual Router Protocol
UDP       9         User Datagram Protocol
VOIP      14        Voice over IP
VRRP      16        Virtual Router Redundancy Protocol
VRRP6     13        Virtual Router Redundancy Protocol for IPv6
WLAN      12        Wireless LAN
WWAN      16        Wireless WAN Interface
X252      23        X.25 Layer 2
X253      27        X.25 Layer 3
```

```
X28       6         X28 Network

ELS+
```

Command history:

Release	Modification
11.00.05	The SMGT event was introduced as of version 11.00.05.
11.01.00	The SMGT event was introduced as of version 11.01.00.

4.4.2.10 NO

Disables the event store system.

Syntax:

```
ELS+no ?
  event-store    Disable event store on flash
ELS+
```

Command History:

Release	Modification
11.00.05	This command was introduced as of version 11.00.05.
11.01.01	This command was introduced as of version 11.01.01.

NO EVENT-STORE

When this command is executed, a warning message appears to alert the user about the potential consequences of running the command.

```
ELS+no event-store
Event store disabled
ELS+
```

4.4.2.11 NVRLOG

Allows you to view the logs stored in the non-volatile memory and to initialize the log.

Syntax:

```
ELS+nvrlog ?
  clear    Initialize the bugs system, deleting all previous ones
  list     Select the number of logs to be displayed on the console
ELS+
```

NVRLOG LIST

Allows you to choose the number of logs that are displayed on the console screen.

Syntax:

```
ELS+nvrlog list <number_of_logs>
```

Example:

```
ELS+nvrlog list 2
 02/26/07 11:31:05 -1- RESET:(CODE 0xc0000000)  EH ES
 BIOS CODE VERSION: 01.09.09  START FROM FLASH L1
 02/26/07 11:28:27 -3- Reload issued by the user
ELS+
```

For each log, you are shown the date and time the event took place, the associated priority level of the logged message (between dashes) and a text with information about the event.

NVRLOG CLEAR

Initializes the log, deleting all previous ones.

Example:

```
ELS+NVRLOG CLEAR 0
```

```
 01/02/07 10:32:47 -1- Logging memory initialized.
ELS+
```

4.4.2.12 RESTORE-CONFIGURATION

Restores the event configuration that the device had on startup.

Syntax:

```
ELS+restore
```

Example:

```
ELS+restore
Do you want to restore ELS configuration?(Y/N)(N): y
ELS+
```

4.4.2.13 SHOW-STORED-LOG

Allows you to see the last event log generated by the **event-store** command if the OS crashes. Events are displayed from most recent to least recent.

Syntax:

```
ELS+show-stored-log ?
  <1..10000>    Show a specific number of events stored
  <cr>
ELS+
```

The system allows you to view a specified number of stored events or all of the events. If you choose the option to view all of the events and you find that there are too many, type Ctrl+P to stop the command.

Example:

```
ELS+show-stored-log 2
Press (Control + P) if you want to finish operation
04/14/16 17:55:27  ethernet0/0: PHY: Link UP - 100Mbps - Full - Copper
04/14/16 17:55:26  ethernet0/0: PHY is Marvell 88E1011S (01410c67)
ELS+
```

```
ELS+show-stored-log
Press (Control + P) if you want to finish operation
04/14/16 17:55:27  ethernet0/0: PHY: Link UP - 100Mbps - Full - Copper
04/14/16 17:55:26  ethernet0/0: PHY is Marvell 88E1011S (01410c67)
04/14/16 17:55:26  GW.005 Bffrs: 25575 avail 25575 idle   fair 1312 low 5063
04/14/16 17:55:26  GW.073 CIT Ver Chg: Ver: 11.01.01-Beta-f97b83d+
04/14/16 17:55:26  GW.002 Portable CGW ATLAS160Super Rel 11.01.01-Beta-f97b83d+ strtd
04/14/16 17:55:26  GW.001 System restarted -- ATLAS160Super router cold start
ELS+
```

Command History:

Release	Modification
11.00.05	This command was introduced as of version 11.00.05.
11.01.01	This command was introduced as of version 11.01.01.

4.4.2.14 VIEW

Allows events to be displayed in the active process.

This command is available from any process, even though it does not appear when requesting all available commands. It must be written in full.

Syntax:

```
ELS+view ?
  history    Display the events history
  <cr>
ELS+view
```

Example:

```
ELS+view
ELS+
```

4.4.2.14.1 VIEW HISTORY

Accesses the history of displayed events.

This history stores the same number of events configured with the **evbuffer** command, as long as it is greater than 1000. If it is lower, then 1000 are stored.

Syntax:

```
ELS+view history ?
  flush    Delete the events history
  regex    Apply a filter to the events history
  <cr>
ELS+view history
```

Example:

```
ELS+view history
ELS+
```

VIEW HISTORY FLUSH

Deletes the history of displayed events.

Example:

```
ELS+view history flush
ELS+
```

VIEW HISTORY REGEX

Applies a filter, in the form of a regular expression, to the history of displayed events.

Syntax:

```
ELS+view history regex ?
  <0..40 chars>    regex
```

Example 1:

A filter is specified so that only ARP historical events are shown.

```
ELS+view history regex ARP\.
  22155610.993295ms  ARP.002 Pkt in 1 1 800 ethernet0/0
  22155611.017757ms  ARP.008 rcv IP RQST 192.168.212.103->192.168.212.96 ifc ethernet0/0
  22156151.921871ms  ARP.002 Pkt in 1 1 800 ethernet0/0
  22156151.945174ms  ARP.008 rcv IP RQST 192.168.212.103->192.168.212.96 ifc ethernet0/0
  22156691.510117ms  ARP.002 Pkt in 1 1 800 ethernet0/0
  22156691.533632ms  ARP.008 rcv IP RQST 192.168.212.103->192.168.212.96 ifc ethernet0/0
  22157237.364754ms  ARP.002 Pkt in 1 1 800 ethernet0/0
  22157237.387300ms  ARP.008 rcv IP RQST 192.168.212.103->192.168.212.96 ifc ethernet0/0
ELS+
```

Example 2:

A filter is specified so that only the historical events containing the **celluar1/1** text are displayed.

```
ELS+view history regex cellular1/1
  22156053.083756ms  GW.019 Slf tst ifc cellular1/1
  22156053.131998ms  GW.022 Nt fld slf tst ifc cellular1/1
  22157053.106362ms  GW.019 Slf tst ifc cellular1/1
  22157053.151817ms  GW.022 Nt fld slf tst ifc cellular1/1
  22158053.135392ms  GW.019 Slf tst ifc cellular1/1
ELS+
```

4.4.2.15 EXIT

Exits Event Logging System monitoring and returns to the plus (+) prompt.

Syntax:

```
ELS+exit
```

Syntax:

```
ELS+exit
+
```

4.5 Supported customizable parameters

Teldat devices have customizable parameters that can modify device behavior *under special circumstances* (customized versions). For more information about enabling, disabling and listing these parameters, please see the **help** command in the **enable patch, disable patch** and **list patch** commands, respectively. These can be found under *Teldat Router Configuration* on page 19.

The Event Logging System has the following customizable parameters:

SRE_INT_FLAGS

This patch lets you modify Event Logging system behavior. Normally, you won't need to use this command, but it is a useful parameter to have for debugging and development. The parameter value is the sum of the flags used.

Ve: 0 The Event Logging System is operating normally.

Flag: 1 The Event Logging System stops the registering of events in the circular display buffer when it is full. This way, when the events are not displayed and the buffer is full, CPU time is not consumed and the system can increase its performance. The side effect is that the most recent events (not the oldest) are lost.

Flag: 2 The date and time are not displayed, so the real-time clock is not queried and performance improves.

Example:

```
Config>enable patch sre_int_flags 1
Config>list patch
Patch Name                    Value
-----------------------------------------
SRE_INT_FLAGS                     1 (0x1)
```