

Technical Documentation



**VIRTEL Audit and Performance**

**User's Guide**

Version : 4.56

Release Date : 1 Sep 2016 Publication Date : 18/01/2017

Syspertec Communication

196, Bureaux de la Colline 92213 Saint-Cloud Cedex Tél. : +33 (0)1 46 02 60 42

[www.syspertec.com](http://www.syspertec.com/)

NOTICE

Reproduction, transfer, distribution, or storage, in any form, of all or any part of the contents of this document, except by prior authorization of SysperTec Communication, is prohibited.

Every possible effort has been made by SysperTec Communication to ensure that this document is complete and relevant. In no case can SysperTec Communication be held responsible for any damages, direct or indirect, caused by errors or omissions in this document.

As SysperTec Communication uses a continuous development methodology; the information contained in this document may be subject to change without notice. Nothing in this document should be construed in any manner as conferring a right to use, in whole or in part, the products or trademarks quoted herein.

"SysperTec Communication" and "VIRTEL" are registered trademarks. Names of other products and companies mentioned in this document may be trademarks or registered trademarks of their respective owners.

**Table of contents**

## 1.

**2.**

### 2.1.

**2.2.**

**2.3.**

**2.4.**

**2.5.**

*2.5.1.*

*2.5.2.*

*2.5.3.*

*2.5.4.*

## 3.

### 3.1.

**3.2.**

**3.3.**

**3.4.**

**3.5.**

*3.5.1.*

*3.5.2.*

*3.5.3.*

## 4.

### 4.1.

**4.2.**

**5.**

**5.1.**

*5.1.1.*

*5.1.2.*

*5.1.3.*

### 5.2.

**5.3.**

*5.3.1.*

### 5.4.

[Monitoring lines and virtual circuits 6](#_bookmark1)

[Displaying line status 7](#_bookmark3)

[Access to the application 7](#_bookmark5)

[Security 7](#_bookmark6)

[Objectives 7](#_bookmark7)

[Contents of each field 8](#_bookmark8)

[Associated functions 8](#_bookmark9)

[*Positioning the list* *8*](#_bookmark10)

[*Displaying details of a line* *9*](#_bookmark11)

[*Commands* *9*](#_bookmark12)

[*Return to the configuration menu* *9*](#_bookmark13)

[Displaying virtual circuits 10](#_bookmark14)

[Access to the application 10](#_bookmark16)

[Security 10](#_bookmark17)

[Objectives 10](#_bookmark18)

[Contents of each field 11](#_bookmark19)

[Associated functions 12](#_bookmark20)

[*Positioning the list* *12*](#_bookmark21)

[*Displaying other lines* *12*](#_bookmark22)

[*Return to previous screen* *12*](#_bookmark23)

[How to issue a VIRTEL command 13](#_bookmark24)

[MVS environment 13](#_bookmark26)

[VSE environment 13](#_bookmark27)

[List of commands 15](#_bookmark28)

[Displaying VIRTEL lines and terminals 15](#_bookmark29)

[*List of lines* *15*](#_bookmark30)

[*List of terminals associated with a line* *15*](#_bookmark32)

[*List of relays* *15*](#_bookmark34)

[Starting and stopping a line 15](#_bookmark36)

[Displaying Memory 16](#_bookmark38)

[*Management* *17*](#_bookmark40)

[Stopping VIRTEL 17](#_bookmark41)

**5.5.**

**5.6.**

*5.6.1.*

*5.6.2.*

*5.6.3.*

*5.6.4.*

*5.6.5.*

### 5.7.

*5.7.1.*

*5.7.2.*

*5.7.3.*

*5.7.4.*

### 5.8.

**5.9.**

**5.10.**

**5.11.**

**5.12.**

**5.13.**

**6.**

**6.1.**

**6.2.**

*6.2.1.*

### 6.3.

*6.3.1.*

*6.3.2.*

### 6.4.

*6.4.1.*

### 6.5.

*6.5.1.*

*6.5.2.*

### 6.6.

*6.6.1.*

*6.6.2.*

## 7.

### 7.1.

*7.1.1.*

*7.1.2.*

*7.1.3.*

*7.1.4.*

*7.1.5.*

*7.1.6.*

*7.1.7.*

*7.1.8.*

*7.1.9.*

### [Stopping a scenario 17](#_bookmark43)

### [Activating and deactivating a terminal or line trace 18](#_bookmark45)

[*Terminal trace* *18*](#_bookmark46)

[*Line trace* *18*](#_bookmark48)

[*Alternate forms of trace commands* *18*](#_bookmark50)

[*Display list of active traces* *19*](#_bookmark52)

[*Deactivate all traces* *19*](#_bookmark54)

### [Obtaining a SNAP 19](#_bookmark56)

[*System level SNAP* *19*](#_bookmark58)

[*Terminal or Relay level SNAP* *19*](#_bookmark60)

[*Message-triggered SNAP* *19*](#_bookmark62)

1. [*column SNAP* *20*](#_bookmark64)

### [Adjusting the SNAP format 20](#_bookmark66)

### [Refreshing a VIRTEL program 20](#_bookmark68)

### [Refreshing a VIRSV service program 20](#_bookmark70)

### [Sending a message to VIRTEL multi-session users 21](#_bookmark72)

### [Suppressing connection and disconnection messages 21](#_bookmark74)

### [Patching a VIRTEL program 21](#_bookmark76)

## [VIRLOG, TRACE, SNAP 22](#_bookmark78)

### [Introduction 22](#_bookmark79)

### [The CONSOLE file 22](#_bookmark80)

[*Example of CONSOLE file* *22*](#_bookmark81)

### [The VIRLOG file 23](#_bookmark82)

[*Example of VIRLOG (HTTP)* *23*](#_bookmark83)

[*Example of VIRLOG (X25)* *24*](#_bookmark84)

### [The VIRTEL LOGGER file 25](#_bookmark85)

[*Extracting and formating the VIRTEL LOG* *25*](#_bookmark86)

### [VIRTEL traces 25](#_bookmark87)

[*Contents of the trace* *26*](#_bookmark89)

[*Examples of traces* *26*](#_bookmark90)

### [VIRTEL SNAP 28](#_bookmark91)

[*Example of SNAP* *28*](#_bookmark93)

[*Message-triggered SNAPMSG* *29*](#_bookmark94)

## [Statistics 30](#_bookmark95)

### [The VIRSTAT file 30](#_bookmark96)

[*VIRSTAT classic format* *30*](#_bookmark98)

[*VIRSTAT alternate X25 format* *31*](#_bookmark99)

[*VIRSTAT formats for VIRTEL Web Access* *32*](#_bookmark100)

[*Statistics file management* *34*](#_bookmark101)

[*Printing the contents of the VIRSTAT file (X25)* *34*](#_bookmark105)

[*Printing the contents of the VIRSTAT file (Web)* *35*](#_bookmark107)

[*SMF Support* *40*](#_bookmark109)

[*Printing the contents of the VIRSTAT SMF records* *40*](#_bookmark111)

[*Messages* *41*](#_bookmark113)

## 8.

### 8.1.

*8.1.1.*

*8.1.2.*

*8.1.3.*

*8.1.4.*

### 8.2.

**9.**

**9.1.**

*9.1.1.*

*9.1.2.*

*9.1.3.*

### 9.2.

**9.3.**

**9.4.**

**10.**

**10.1.**

## [Memory management 42](#_bookmark114)

### [Access to the application 42](#_bookmark116)

[*Memory display in MEMORY=TEST mode* *43*](#_bookmark118)

[*Positioning the display* *43*](#_bookmark119)

[*Real time monitoring* *44*](#_bookmark120)

[*Return to the sub-application menu* *44*](#_bookmark121)

### [Virtel Memory Display 44](#_bookmark122)

## [Memory trace management 45](#_bookmark123)

### [Memory trace commands 45](#_bookmark124)

[*Activating memory trace* *45*](#_bookmark125)

[*Reseting memory trace* *45*](#_bookmark127)

[*Stopping memory trace* *46*](#_bookmark128)

### [Memory trace from the VIRTCT 46](#_bookmark129)

### [Memory trace analysis 46](#_bookmark130)

### [Memory trace overhead 47](#_bookmark131)

## [Trademarks 48](#_bookmark132)

### [Open source software 48](#_bookmark133)

# Monitoring lines and virtual circuits

The Line Status sub-application allows the administrator to display the current status of lines and virtual circuits (CVC) managed by VIRTEL control, and optionally to modify the status of lines.

STATUS of LINES:

---------------- Applid: SPVIRD2 13:12:21

C Name

In Out Links Description

Seen

# Displaying line status

**2.1. Access To The Application**

The Line Status sub-application is invoked by pressing [PF9] in the Configuration Menu, by pressing [PF10] in the Sub- Application Menu, or via the Multi-Session Menu using a transaction which calls module VIR0027.

**2.2. Security**

When the security subsystem is active, access to Line Status sub-application from the Configuration Menu or the Sub- Application Menu is controlled by the resource $$UTIL$$.

When accessed by a transaction, normal transaction security rules will apply.

Security management is described in chapter 4 of the VIRTEL Technical Documentation.

**2.3. Objectives**

This sub-application begins by displaying the Line Status Display screen. Started lines are displayed in high-intensity or white text, stopped lines are displayed in low intensity or blue text.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C-HTTP | 0 | 0 | 26 | HTTP line (entry point CLIWHOST) |
| G-HTTP | 0 | 0 | 26 | Connexions en mode HTTP (GLIWHOST) |
| H-HTTP | 0 | 0 | 32 | HTTP line (entry point DEMOHTTP) |
| S-SMTP | 0 | 0 | 16 | [client.com<virtel@client.com>](mailto:virtel@client.com) |
| W-HTTP | 0 | 0 | 26 | HTTP line (entry point WEB2HOST) |

*Line Status Display screen*

**2.4. Contents Of Each Field**

### STATUS OF-LINES

Allows the administrator to display a subset of lines, by typing the first character of the name of each desired line into this field and pressing [Enter]. If the field is blank, all lines are displayed.

### C

Command input field.

### Name

The internal name of the line.

### In

The number of virtual circuits currently in use by incoming calls.

### Out

The number of virtual circuits currently in use by outgoing calls.

### Links

The number of terminals linked to the line.

### Description

Comments.

### Seen

User name.

**2.5. Associated Functions**

**2.5.1. Positioning the list**

If the line status display occupies more than one screen, you can scroll through the list of lines by using [PF5], [PF7] and [PF8].

### [PF5]

return to the first page of the list.

### [PF7]

scroll back to previous page.

### [PF8]

scroll forward to next page.

**2.5.2. Displaying details of a line**

To display information about the virtual circuits linked to a VIRTEL line, place the cursor on the line required and press [PF12].

**2.5.3. Commands**

To send a command to a line, place the cursor in the “C” field in front of the line name, type the command, then press [Enter]. The commands available are:

### s

starts a line. If the line is already started, VIRTEL attempts to start or restart any terminals associated with the line but not currently linked. This allows VIRTEL to recover LU’s which have been deactivated and reactivated by VTAM, without stopping the line.

### p

stops a line.

The LINE START and STOP commands can also be issued from the MVS or VSE console. Se[e “Starting and stopping a](#_bookmark36) [line”, page 15](#_bookmark36).

**2.5.4. Return to the configuration menu**

To return to the configuration menu, press [PF3] or [Clear].

ACTIVE TERMINALS for LINE: H-HTTP ---------------- Applid: SPVIRD2 14:34:36

# Displaying virtual circuits

**3.1. Access To The Application**

To display the status of the virtual circuits associated with a line, place the cursor on the desired line in the Line Status Display screen and press [PF12].

**3.2. Security**

Security rules are the same as those which apply to the previous screen.

**3.3. Objectives**

This sub-application begins by displaying the Virtual Circuit Display screen for the selected line, as shown in the example below:

|  |  |  |  |
| --- | --- | --- | --- |
| Prefix : HT Type : TCP1 |  | Defined : 32 Linked : | 32 |
| Number of occupied circuits : | 3 | Number of connections : | 38 |
| Maximum simultaneously used : | 4 | Total time connected : | 15 mn |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Terminal | User | Sends | Time | Node | Remote number | Call Data |
| HTVTA003 | SPUSERA | 5 | 5 mn | RWTVT000 | 192.168.000.013 | DEMOHTTP |
| HTVTA001 | SPUSERE | 5 | 4 mn | RWTVT002 | 192.168.000.046 | DEMOHTTP |
| HTVTA000 | SPUSERD | 5 | 4 mn | RWTVT003 | 192.168.000.025 | DEMOHTTP |

P3=Return

P4=Next Line

P5=First Line

P7=Previous P8=Next

*Virtual Circuit Display screen*

**3.4. Contents Of Each Field**

### ACTIVE TERMINALS for LINE

Indicates the internal name of the line whose virtual circuits are being displayed.

### Prefix

The terminal name prefix associated with this line.

### Type

The line type, as defined in the line definition.

### Defined

The number of terminals defined for this line.

### Linked

The number of terminals currently linked to this line.

### Number of occupied circuits

The number of virtual circuits currently in use.

### Number of connections

The total number of calls received.

### Maximum simultaneously used

The maximum number of virtual circuits in use at any one time.

### Total time connected

The total connection time.

### Terminal

The terminal name (name of the virtual circuit).

### User

User name if signed on to VIRTEL.

### Sends

The number of messages sent to the terminal.

### Time

The connection time in minutes.

### Node (for Minitel)

The name of the node to which the terminal is currently connected.

### Node (for HTTP lines)

The relay name (3270 LU name) used to connect to the host application.

### Remote number (for X25 lines)

The X25 called number for an outgoing call, or the X25 calling number for an incoming call.

### Remote number (for HTTP lines)

The IP address of the client.

### Call Data (for X25 lines)

The call user data field of the call packet (for both incoming and outgoing calls).

### Call Data (for HTTP lines)

The external name of the transaction which represents the directory (pathname) in the URL.

**3.5. Associated Functions**

**3.5.1. Positioning the list**

If the Virtual Circuit Status Display occupies more than one screen, you can scroll through the list of terminals by using [PF7] and [PF8].

### [PF7]

scroll back to previous page.

### [PF8]

scroll forward to next page.

**3.5.2. Displaying other lines**

You can use the [PF4] and [PF5] keys to display information about the other lines under VIRTEL control. To view the Virtual Circuit Status Display screen for the following line, press [PF4]. To return to the Virtual Circuit Status Display screen for the first line defined in VIRTEL, press [PF5].

**3.5.3. Return to previous screen**

To return to the Lines Status Display, press [PF3]. To return to the Configuration Menu, press [Clear].

# How to issue a VIRTEL

**command**

VIRTEL allows certain functions to be controlled dynamically by console commands.

Use one of the following methods to send a command to VIRTEL, according to the operating system:

**4.1. MVS Environment**

The following command may be issued at the z/OS operator console, or from an SDSF session under TSO, in which case the command must be prefixed by the character “/”:

F stcvirte,virtel-cmd

### stcvirte

the name of the VIRTEL started task STC

### virtel-cmd

a VIRTEL command, as described in the following section

**4.2. VSE Environment**

To send a command to VIRTEL, issue the following command at the VSE operator console:

MSG virtel,DATA=virtel-cmd

### virtel

the VIRTEL jobname (usually VIRTEL), or the partition in which VIRTEL is executing (for example, F4)

### virtel-cmd

a VIRTEL command, as described in the following section Alternatively, issue the following command at the VSE operator console:

MSG Fx

### Fx

Partition in which VIRTEL is executing

The system responds with:

AR 0015 1I40I READY

Fx-nnnn

Note the reply number (nnnn) and issue the following command:

nnnn virtel-cmd

### nnnn

reply number

### virtel-cmd

a VIRTEL command, as described in the following section

# List of commands

**5.1. Displaying VIRTEL Lines And Terminals**

**5.1.1. List of lines**

LINES LINES,ACT LINES,INACT

The LINES command displays the VIRTEL ACB name and a list of the lines defined in the VIRTEL configuration file. The optional keywords ACT or INACT may be used to restrict the display to lines in “started” or “stopped” state respectively.

**5.1.2. List of terminals associated with a line**

LINE=linename,DISPLAY (or L=linename,D)

### linename

internal or external name of the line

The LINE DISPLAY command displays the status of a line and its associated terminals.

**5.1.3. List of relays**

RELAYS

The RELAYS command displays the VIRTEL ACB name and a list of the relay LUs opened by VIRTEL.

**5.2. Starting And Stopping A Line**

LINE=linename,START (or L=linename,S) LINE=linename,STOP (or L=linename,P)

### linename

internal or external name of the line

The LINE START and LINE STOP commands perform the same function as the [“S” and “P” commands](#_bookmark12) on the “Status of lines”. These commands may only be issued for line types AntiGATE, AntiPCNE, AntiFASTC, and TCP/IP.

**5.3. Displaying Memory**

MEMDISPLAY

With the memory diagnostic tool active the MEMDISPLAY command summarize the VIRTEL subpool active allocated memory.

VIR0200I MEMDISPLAY VIR0271I DISPLAY 978

SP1=00024478 SP2=00001044 SP3=0008E35F SP4=00002F61 00910091 00040009 02380294 000B0011 SP5=000317DC SP6=0004DF73 SP7=00000000 SP8=00000220

**00C504C7** 01370137 00000000 00000002

POOL CONTROL BLOCK. SUBPOOL=1

PAG=00109000 NFQ=00109008 #FQ=00000001 FRE=0000B2A0 PAG=000F9000 NFQ=000F9008 #FQ=00000001 FRE=00000080 PAG=000E9000 NFQ=000E9008 #FQ=00000001 FRE=00000078 POOL CONTROL BLOCK. SUBPOOL=2

PAG=1EC14000 NFQ=1EC14008 #FQ=00000005 FRE=0000EF68 POOL CONTROL BLOCK. SUBPOOL=3

PAG=1ECD4000 NFQ=1ECD4008 #FQ=00000002 FRE=00009DF8 PAG=1ED54000 NFQ=1ED54008 #FQ=00000002 FRE=00001750 PAG=1EC84000 NFQ=1EC84008 #FQ=00000001 FRE=00000878 PAG=1ED04000 NFQ=1ED04008 #FQ=00000001 FRE=00000878 PAG=1ED94000 NFQ=1ED94008 #FQ=00000002 FRE=00002768 PAG=1ECF4000 NFQ=1ECF4008 #FQ=00000001 FRE=00000878 PAG=1EE04000 NFQ=1EE04008 #FQ=00000001 FRE=00000878 PAG=1ED74000 NFQ=1ED74008 #FQ=00000001 FRE=00000878 PAG=1ECE4000 NFQ=1ECE4008 #FQ=00000001 FRE=00000878 PAG=1EC64000 NFQ=1EC64008 #FQ=00000001 FRE=00000878 POOL CONTROL BLOCK. SUBPOOL=4

PAG=1EC04000 NFQ=1EC04008 #FQ=00000004 FRE=0000CFA0 POOL CONTROL BLOCK. SUBPOOL=5

PAG=1ECA4000 NFQ=1ECA4008 #FQ=00000002 FRE=0000D870 PAG=1ED14000 NFQ=1ED14008 #FQ=00000001 FRE=000043B8 PAG=1ED24000 NFQ=1ED24008 #FQ=00000001 FRE=000043B8 PAG=1EC74000 NFQ=1EC74008 #FQ=00000001 FRE=0000A1D8 PAG=1EC54000 NFQ=1EC54008 #FQ=00000001 FRE=0000A1D8 PAG=1EBB4000 NFQ=1EBB4008 #FQ=00000001 FRE=000043B8 POOL CONTROL BLOCK. SUBPOOL=6

PAG=1EBF4000 NFQ=1EBF4008 #FQ=00000002 FRE=00000A50 PAG=1EBE4000 NFQ=1EBE4008 #FQ=00000001 FRE=00000088 PAG=1EBD4000 NFQ=1EBD4008 #FQ=00000001 FRE=000000B8 PAG=1EBC4000 NFQ=1EBC4008 #FQ=00000001 FRE=000000D0 PAG=1EBA4000 NFQ=1EBA4008 #FQ=00000001 FRE=00000108 POOL CONTROL BLOCK. SUBPOOL=7

PAG=00000000 NFQ=00000000 #FQ=00000000 FRE=00000000 POOL CONTROL BLOCK. SUBPOOL=8

PAG=1ECB4000 NFQ=1ECB4008 #FQ=00000001 FRE=0000FDB0 ALLOC=0001240K, FREE=0000479K, TOTAL=0001728K VIR0272I END

The display response is split into a summary section for each subpool and a detailed allocated page block and free queue element display for each subpool.

In the summary display, each subpool has two displayed values. The top value is the amount of storage currently allocated and the value below represents the current allocation in 1K chunks and a peak allocation in 1K chunks.

For example in the above display in SP5 we can see that there is an allocated value of 317DC bytes, represented by 00C5 in 1K chunks, and a peak value of 04C7 in 1K chunks.

At the bottom of the display is a line which provide allocated, free and total values.

**5.3.1. Management**

Memory Display feature is activated by using the MEMHST sub parameter in the MEMORY parameter present in the VIRTCT.(see the VIRTCT subparameter MEMHST in “VIRTEL456 Installation User Guide”).

It can be deactivated by using the command:

F VIRTEL,MEMDISPLAY,DISABLE

This should only be implemented when advised to do so by Technical Support. Performance degradation might occur due to the additional monitoring services. This will depend on VIRTEL demand.

**5.4. Stopping VIRTEL**

STOP

The STOP command allows to STOP the VIRTEL task. This command is intended to be mainly used in VSE environment even if it is also available in MVS environmment. On MVS environment you can also use the following command :

P stcvirte

### stcvirte

the name of the VIRTEL started task STC

**5.5. Stopping A Scenario**

KILL,T=termid

### termid

terminal name

The KILL command requests VIRTEL to abnormally terminate the scenario currently active on the specified terminal.

**5.6. Activating And Deactivating A Terminal Or Line Trace**

**5.6.1. Terminal trace**

A trace can be activated on the device or on his relay.

TERM=termid,TRACE (or T=termid,T) TERM=termid,NOTRACE (or T=termid,N) RELAY=relayname,TRACE (or R=relayname,T) RELAY=relayname,NOTRACE (or R=relayname,N)

### termid

terminal name

### relayname

relay associated to the terminal

It is often easier to identify the relay used whose name appears at the bottom of the 3270 session screen as shown below.



Relay and Printer name identification

**5.6.2. Line trace**

LINE=linename,TRACE (or L=linename,T) LINE=linename,NOTRACE (or L=linename,N)

### linename

internal or external name of the line

**5.6.3. Alternate forms of trace commands**

The following alternate forms of the TRACE/NOTRACE commands are also valid

TRACE,T=termid TRACE,L=linename TRACE,R=relayname NOTRACE,T=termid NOTRACE,L=linename NOTRACE,R=relayname

### termid

terminal name

### linename

internal or external name of the line

### relayname

name of VTAM relay LU currently associated with the terminal

**5.6.4. Display list of active traces**

TRACE,DISPLAY (or TRACE,D)

**5.6.5. Deactivate all traces**

NOTRACE,ALL

This command does not affect any memory trace. To stop a memory trace, refer to [“Memory trace management”,](#_bookmark123) [page 45](#_bookmark123).

Se[e “VIRTEL traces”, page 25](#_bookmark87).

**5.7. Obtaining A SNAP**

The SNAP command prints the contents of the VIRTEL internal trace table. Se[e “VIRTEL SNAP”, page 28](#_bookmark91).

**5.7.1. System level SNAP**

SNAP

**5.7.2. Terminal or Relay level SNAP**

SNAP,T=termid SNAP,R=relayname

### termid

terminal name

### relayname

name of VTAM relay LU currently associated with the terminal

**5.7.3. Message-triggered SNAP**

SNAPMSG,ALL

The SNAPMSG command requests VIRTEL to generate an automatic SNAP after certain messages (VIRI902W VIR0026W VIR0052I VIR1552I VIR0526W VIR1952I).

SNAPMSG=(message,search,action)

The SNAPMSG parameter allows a SNAP or DUMP to be taken whenever a particular message number is issued by VIRTEL. The command has an additional search field which can be used to identify a message with a particular character string, for example a specific return code. This feature is also avalable by using the SNAPMSG command from the console. (see “SNAPMSG command” in the VIRTEL Audit and Performance Reference manual).

### message

Any message that can be issued by Virtel.

### search

Any seache criteria issued within the message. The search file is restricted to a maximu of 10 characters. Anything beyon will be ignored. Default search is none.

### action

Possible values are S for SNAP or A for ABEND. Virtel will abend with a U0999 abend code, reason code 15 if the ABEND action is used. Default action is SNAP.

**5.7.4. 80-column SNAP**

SNAP80

The SNAP80 command prints the contents of the VIRTEL internal trace table in 80 column format, whatever the current value of the SNAPW parameter.

**5.8. Adjusting The SNAP Format**

SNAPW=80 ou SNAPW=132

The SNAPW command sets the width for future SNAP commands (80 or 132 columns). The SNAPW parameter in the VIRTCT determines the default width at VIRTEL startup. Refer to the section “Parameters of the VIRTCT” in the VIRTEL Installation Guide for details of the SNAPW parameter.

**5.9. Refreshing A VIRTEL Program**

NEW=progname

### progname

program name

The NEW command requests VIRTEL to load a fresh copy of a program (presentation module, exit, etc) into the VIRTEL address space. This is required after an update has been made to a program. The message VIR0060W PROGRAM progname IS A NEW COPY indicates a successful reload. The message VIR0061W PROGRAM progname NOT IN MEMORY indicates that the program has not yet been loaded into the VIRTEL address space. In this case, VIRTEL will load the program automatically when it is next needed.

**5.10. Refreshing A VIRSV Service Program**

VIRSV,NEW=servname

### servname

service name

The VIRSV,NEW command requests VIRTEL to stop the requested VIRSV service. This has the effect of loading a fresh copy of the associated service program the next time the service is invoked by a scenario. The message VIR0260W SERVICE servname IS A NEW COPY indicates that the service was stopped successfully. The message VIR0261W

SERVICE servname NOT IN MEMORY indicates that the service is not yet started. In this case, VIRTEL will start the service and load the program automatically when it is next needed.

**5.11. Sending A Message To VIRTEL Multi-session Users**

MSG=message text

The specified message will be displayed on the VIRTEL multi-session screen.

**5.12. Suppressing Connection And Disconnection Messages**

SILENCE

The SILENCE command reverses the state of the SILENCE parameter in the VIRTCT. Its purpose is to activate or deactivate the suppression of terminal connection and disconnection messages written to the operator console.

(Refer to the section SILENCE parameter in the "Parameters of the VIRTCT" in the VIRTEL Installation Guide for a list messages affected by this command.)

**5.13. Patching A VIRTEL Program**

ZAP=progname+offset,verify,replace

### progname

program name

### offset

offset into program

### verify

verify value (2 to 8 hexadecimal digits)

### replace

replacement value (2 to 8 hexadecimal digits)

The ZAP command allows the dynamic application of a corrective patch to a program while VIRTEL is running. This command is intended to be used only under the advice of Syspertec technical support personnel.

# VIRLOG, TRACE, SNAP

**6.1. Introduction**

The VIRTEL started task offers the administrator 5 sources of information to verify the correct functioning of VIRTEL, to monitor its activity, or to diagnose possible problems:

* the CONSOLE file
* the VIRLOG file
* the VIRTEL Logger
* the TRACE in the VIRTRACE file
* the SNAP in the SYSPRINT file

**6.2. The CONSOLE File**

In **MVS environment**, the CONSOLE file is written to the VIRTEL started task’s JESMSGLG file.

In **VSE environment**, the CONSOLE file is written to the VIRTEL partition’s POWER LST file (LISTLOG)

The CONSOLE file allows the administrator to monitor the startup and subsequent activity of VIRTEL. Using the console file, the administrator can check that the VSAM files are correctly opened, verify that the customer key has been correctly recognized, check the initialization of the TCP/IP sockets interface using the correct IP address and port, and monitor connections and disconnections of terminals and applications.

The SILENCE=YES parameter in the VIRTCT allows the suppression of certain console messages relating to the connection and disconnection of terminals.

**6.2.1. Example of CONSOLE file**

J E S 2 J O B L O G -- S Y S T E M M 2 3 5 -- N O D E N 1 10.10.17 STC07142 ---- MONDAY, 27 FEB 2006 ----

10.10.17 STC07142 VARY NET,ACT,ID=APPLVIRT

10.10.17 STC07142 IEF695I START SPVIRBW WITH JOBNAME SPVIRBW IS ASSIGNED TO USER VIRTEL, GROUP TPPROD

10.10.17 STC07142 $HASP373 SPVIRBW STARTED

10.10.17 STC07142 IEF403I SPVIRBW - STARTED - TIME=10.10.17

10.10.19 STC07142 +VIR0000I STARTING LICENCE P500 - PERMANENT (2999 - 12 - 31)

* + 1. STC07142 +VIR0019I VIRTEL 4.32 HAS NO PTFS APPLIED
    2. STC07142 +VIR0024I OPENING FILE VIRARBO
    3. STC07142 +VIR0024I OPENING FILE VIRSWAP
    4. STC07142 +VIR0024I OPENING FILE VIRCMP3

10.10.21 STC07142 +VIR0024I OPENING FILE VIRCAPT

* + 1. STC07142 +VIR0024I OPENING FILE VIRHTML
    2. STC07142 +VIR0024I OPENING FILE PCHOST1

10.10.22 STC07142 +VIR0024I OPENING FILE HTMLTRSF

10.10.22 STC07142 +VIR0024I ATTACHING SUBTASKS

* + 1. STC07142 +VIR0604I VIRSTAT NOW RECORDING ON VIRSTATA DSN=SP000.SPVIRBW.STATA
    2. STC07142 +VIR0024I READING VIRARBO

10.10.23 STC07142 +VIR0035E UNDEFINED LINE ADMRSET1 FOR RULE UPLOAD1A

10.10.23 STC07142 +VIR0035E UNDEFINED LINE ADMRSET1 FOR RULE UPLOAD1B

10.10.23 STC07142 +VIR0005W UNABLE TO ACTIVATE RHTIM000 (HTIMP000) ERROR: 58000000

10.10.23 STC07142 +VIR0024I READING TYPES

10.10.23 STC07142 +VIR0027I 0 SCREEN TYPES LOADED USING 0K

10.10.23 STC07142 +VIR0000I THIS COPY OF VIRTEL IS FOR THE EXCLUSIVE USE OF:

10.10.23 STC07142 +VIR0000I SYSPERTEC COMMUNICATION

10.10.23 STC07142 +VIR0000I 196 BUREAUX DE LA COLLINE

* + 1. STC07142 +VIR0000I 92213 SAINT CLOUD CEDEX

10.10.23 STC07142 +VIR0000I HTTP Date: Mon, 27 Feb 2006 07:10:23 GMT 10.10.23 STC07142 +VIR0000I SMTP Date: Mon, 27 Feb 2006 08:10:23 +0100

10.10.23 STC07142 +VIR0000I SPVIRBW STARTED AT 27/02/06 10:10:23 , VERSION 4.32

* + 1. STC07142 +VIRT903W LINE HTTP-LIG HAS A SESSION STARTED WITH TCP/IP TCPIP

10.10.24 STC07142 +VIRHT01I HTTP INITIALISATION FOR HTTP-LIG (H-HTTP ), VERSION 4.32 10.10.24 STC07142 +VIRT912W HTTP-LIG SOCKET 00000000 STARTED FOR 192.168.235.030:41000

10.10.24 STC07142 +VIRT903W LINE SMTP-LIG HAS A SESSION STARTED WITH TCP/IP TCPIP

10.10.24 STC07142 +VIRSM01I SMTP INITIALISATION FOR SMTP-LIG (S-SMTP ), VERSION 4.32 10.10.24 STC07142 +VIRT912W SMTP-LIG SOCKET 00000000 STARTED FOR 192.168.235.030:42000

10.10.24 STC07142 +VIRT903W LINE HTTP-W2H HAS A SESSION STARTED WITH TCP/IP TCPIP

* + 1. STC07142 +VIRHT01I HTTP INITIALISATION FOR HTTP-W2H (W-HTTP ), VERSION 4.32 10.10.24 STC07142 +VIRT912W HTTP-W2H SOCKET 00000000 STARTED FOR 192.168.235.030:41001
    2. STC07142 +VIR0505I LINKING TERMINAL HTLOC000 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC001 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC002 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC003 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC004 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC005 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC006 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC007 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC008 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC009 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC010 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC011 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC012 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC013 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC014 TO H-HTTP

10.10.25 STC07142 +VIR0505I LINKING TERMINAL HTLOC015 TO H-HTTP

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA000 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA001 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA002 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA003 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA004 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA005 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA006 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA007 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA008 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA009 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA010 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA011 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA012 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA013 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA014 TO H-HTTP RELAY \*HTTPOOL

10.10.25 STC07142 +VIR0507I LINKING TERMINAL HTVTA015 TO H-HTTP RELAY \*HTTPOOL

*Example of CONSOLE file*

**6.3. The VIRLOG File**

This is a printable file with record length 131 and record format FA which provides a record of IP connections to VIRTEL.

**6.3.1. Example of VIRLOG (HTTP)**

The figure below shows an example of VIRLOG entries for incoming HTTP calls:

£Software: VIRTEL 4.32

£Date: 02/01/06

£Line W-HTTP W-HTTP W-HTTP W-HTTP W-HTTP

Local

Pseudo

Started Ended

Price

Received Sent

Remote Address User

WHT00200 DELOC003 I 15.34.53 15.34.53 00000007 00000381 00023135 192.168.000.043

WHT00200 DELOC002 I 15.34.53 15.34.53 00000004 00000381 00010833 192.168.000.043

WHT00200 DELOC003 I 15.34.53 15.34.53 00000007 00000386 00006976 192.168.000.043

WHT00200 DEVTA003 I 15.34.53 15.34.59 00000649 00001169 00010397 192.168.000.043

WHT00200 DELOC002 I 15.35.02 15.35.02 00000005 00000402 00000049 192.168.000.043

200 PUBLIC DATA.JS W2H-DIR

200 PUBLIC JS01.JS W2H-DIR

200 PUBLIC VIRTBLUE W2H-DIR PUBLIC WEB2VIRT W2H-10

304 WEB2HOSTXHTML.JP W2H-DIR

|  |  |  |
| --- | --- | --- |
| W-HTTP | WHT00200 DEVTA003 I 15.35.20 15.35.52 00003275 00008936 00095705 192.168.000.043 SPTBOWL | PUBLIC WEB2VIRT W2H-13 |
| W-HTTP | WHT00200 DELOC003 I 15.35.53 15.35.53 00000002 00000875 00000248 192.168.000.043 | 200 PUBLIC WEB2VIRT W2H-DIR |
| W-HTTP | WHT00200 DELOC003 I 15.35.54 15.35.54 00000002 00000458 00000049 192.168.000.043 | 304 WEB2HOSTWEB2HOST W2H-DIR |

*Example of VIRLOG file (HTTP)*

The LINE column shows the internal name of the HTTP line.

The LOCAL column shows the name of the rule selected for each call. The PSEUDO column shows the VIRTEL terminal name used.

The next column contains “I” to indicate this is an incoming call.

The STARTED and ENDED columns show the start and end time of each IP session.

The PRICE column represents the duration of the transaction in hundredths of a second This value may be modified by exit 7.

The RECEIVED and SENT columns contain the number of bytes received from and sent to the browser. The REMOTE ADDRESS column contains the IP address of the browser.

The USER column contains the userid if the transaction is secured. The next column contains the HTTP status code (for static pages) The last three 8-byte columns represent:

* The external name of the VIRTEL transaction which represents the HTTP path name
* The name of the HTML page
* For static pages: The name of the VIRTEL directory containing the HTML page
* For dynamic pages: The internal name of the HTTP transaction which was used to populate the page

**6.3.2. Example of VIRLOG (X25)**

The figure below shows an example of VIRLOG entries for X25 calls:

£Software: VIRTEL 4.32

£Date: 11/21/07

£Line Local Pseudo

Started Ended

Price

Received Sent

Remote Address User

X001LINE 001880 X001T007 I 13.47.37 13.48.00 00002288 00000392 00000119 191334833

X001LINE G001T004 X001T000 O 13.48.30 13.48.50 00001966 00000001 00000001 191334833001870

X001LINE P001O001 X001T001 O 13.48.49 13.49.20 00003069 00000001 00000001 001870

X001LINE G001T003 X001T002 O 13.49.01 13.49.22 00002147 00000001 00000001 001870

MINITEL

PCNE1

*Example of VIRLOG file (X25)*

The LINE column shows the internal name of the X25 line.

The LOCAL column shows the called subaddress for incoming calls, or the name of the associated AntiGATE or AntiPCNE terminal for outgoing calls.

The PSEUDO column shows the VIRTEL terminal name used.

In the next column “I” indicates an incoming call, “O” indicates an outgoing call. The STARTED and ENDED columns show the start and end time of each call.

The PRICE column represents the duration of the transaction in hundredths of a second, except for calls on Fast Connect lines, where the PRICE column contains the “X25 units sent” value supplied by NPSI. This value may also be modified by exit 7.

The RECEIVED and SENT columns contain the number of bytes received from and sent to the X25 line.

The REMOTE ADDRESS column contains the caller X25 number for incoming calls, or the called X25 number for outgoing calls.

The last column contains the PCNE call user data (if present), otherwise it contains the default entry point name for X25 calls specified by the DEFENTR parameter in the VIRTCT. For GATE calls this column is blank.

**6.4. The VIRTEL LOGGER File**

The VIRTEL log is written to the system logger when LOG=LOGGER is specified in the TCT. VIR0002B is a batch program that can be run to extract the VIRTEL records from the System Logger.

**6.4.1. Extracting and formating the VIRTEL LOG**

The figure below shows an example of JCL to extract and format the VIRTEL LOG entries recorded in the System Logger:

//\*

//\* DESCRIPTION

//\*

//LOGGER PROC P=

//S01 EXEC PGM=VIR0002B,PARM='&P'

//STEPLIB DD DSN=VIRTEL.LOADLIB,DISP=SHR

//VIRLOG DD SYSOUT=\*,DCB=BLKSIZE=25500

// PEND

//S01 EXEC LOGGER,P='DELETE(>2)'

VIRLOG DCB LRECL=255,BLKSIZE=25500,RECFM=VB

*Example of JCL to extract the VIRTEL LOG from the System Logger*

The available JCL parameters are:

(>nnn)

COPY [-------------------]

(fromdate[,todate])

(>nnn)

DELETE [-------------------]

(date)

The date format is yyyyddd.

**6.4.1.1.**

**Examples**

|  |  |  |  |
| --- | --- | --- | --- |
| COPY COPY(>2) COPY(>0) DELETE(>2) | Copy all records  Copy records older than 2 days Copy up to yesterday  Delete records older than 2 days |  | |
| COPY(2015047) | Copy records from 2015.047 |
| COPY(2015047,2015048) | Copy records from 2015.047 thru to 2015.048 | DELETE(2015047) | Delete records prior to 2015.047 |
| COPY(>0),DELETE(>1) | Will copy records from the previous and earlier, |  |  |
|  | and will then delete from 2 days ago leaving about 24 |  | hours of data in the log stream. |

*Example of VIRTEL LOGGER extraction parameter*

**6.5. VIRTEL Traces**

All messages which pass between a terminal and a host application, or all messages received and sent on a line, can be traced to a print file.

Activation and deactivation of a trace on a terminal or a line is performed by means of the TRACE and NOTRACE commands (se[e “VIRTEL commands”, page 1](#_bookmark0) and [“Activating and deactivating a terminal or line trace”, page 18](#_bookmark45)).

A terminal or line trace remains active until a corresponding NOTRACE command is issued or until the VIRTEL started task terminates.

It is also possible to trace specific incoming calls (“tracing by rule”). In this case, activation of the trace is specified in the definition of the rule which VIRTEL uses to route the incoming call. For example, a rule can be created to activate the trace for calls which originate from a specific terminal address (X25 or IP). The trace can be activated for commands and/or data packets.

Activation or deactivation of a “trace by rule” is performed via the VIRTEL on-line configuration menus, and consists of updating the “Trace” field in the rule definition, followed by pressing the F1 key. See “Rules” in the VIRTEL Connectivity Reference manual for more details.

A “trace by rule” remains active as long as the “Trace” field in the rule definition is not empty. Message VIR0036W confirms the activation of the trace.

In **MVS environment**, the trace data is written to the VIRTRACE file in the VIRTEL started task. In **VSE environment**, the trace data is written to the POWER LST file of the VIRTEL partition.

Activation and deactivation of a memory trace is performed by means of the MEMTRACE and NOMEMTRACE commands (se[e “Memory Trace Management”, page 45](#_bookmark123)). The allocation memory is written in the SNAP file when a SNAP command is issued.

**6.5.1. Contents of the trace**

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**6.5.2. Examples of traces**

LCL712 00000 LCL712 00000

00020

00040

00060

00080

000A0

000C0

000E0

00100

00120

11A: from application SPCICST

13:05:47.48

F1C2

\*1B

\* 099A95B4

11A: from application SPCICST

13:05:47.49

F5C2114B E9131140 5B290242 F1C0F8E2 40290242 F4C0F0C1 D7D7D3C9 C4290242

42F4C0F0 E3A89785 40A896A4 9940A4A2

846B40A3 88859540 979985A2 A240C5D5

99898440 4B404B40 4B404B29 0241F442

9996A497 8984404B 404B404B 290241F4 D781A2A2 A6969984 404B404B 404B2903 42F4C0F0 D3819587 A4818785 404B404B

290242F4 C0F0D585 A640D781 A2A2A696

1150F11D F0115A50 1D7C115B 5B1DF011

89879596 9540A396 40C3C9C3 E24011C1 F5C0F0E2 D7C3C9C3 E2E34011 C8F02902 85998984 40819584 409781A2 A2A69699 E3C5D97A 114BD929 0242F4C0 F0E4A285 F5114BF1 1DF0114B F4290242 F4C0F0C7 42F5114C 4B1DF011 4CE92902 42F4C0F0

41F442F5 C04C114D C11DF011 4DF92902

404B2902 41F442F5 114E4C1D F01150D5

9984404B 404B404B 290341F4 42F5C04C

5B602902 42F2C0F8 C4C6C8C3 C5F3F5F2

\*5B..Z.. $...1é8Signon to CICS .A\*

\* ...4é0APPLID...5é0SPCICST .H0..\*

\*.4é0Type your userid and passwor\*

\*d, then press ENTER:..R...4é0Use\*

\*rid . . . ....4.5..1.0..4...4é0G\*

\*roupid . . ....4.5.<..0.<Z...4é0\*

\*Password . . ....4.5é<.(A.0.(9..\*

\*.4é0Language . . ....4.5.+<.0.&N\*

\*...4é0New Password . . ....4.5é<\*

\*.&1.0.&.to.$$.0.$-...2é8DFHCE352\*

099A95B4

099A95D4

099A95F4

099A9614

099A9634

099A9654

099A9674

099A9694

099A96B4

099A96D4

00140

00160

F040D793 8581A285 40A3A897 8540A896 40404040 40404040 40404040 40404040

... SAME AS ABOVE ...

40404040 40404040 40404040 4040401D

40404040 40404040 40404040 40404040

40404040 40404040 40404040 40404040

A49940A4 A2859989 844B4040 40404040 \*0 Please type your userid. 40404040 40404040 40404040 40404040 \*

\* 099A96F4

\* 099A9714

00280

002A0

002C0 LCL712

F8404040 40404040 40404040 40404040

40404040 40404040 40404040 40404040

40404040 40404040 40404040 40404040

\*

\*

\*

.8

\*

\*

\*

099A9834

099A9854

099A9874

11A: to application SPCICST

13:05:51.03

00000 7D4B6C11 4BE9A7A8 A9 \*'.%..Zxyz \* 099A95B4

*Example of terminal trace (inbound 3270 terminal)*

X001T007

XOT: RECEIVED FROM ROUTER

00000 00000019 10010B96 00188019 13348330 X001T007

XOT: SENT TO ROUTER

13:48:15.26

0A420707 43030302 CC0300C4 80

13:48:15.37

\*

....o .....c........... D. \* 0989117C

00000 00000003 10010F \* .... \* 09896178 X001T007 XOT: RECEIVED FROM ROUTER

00000 0000001B 100100D7 C5E2C9E3 404040E2 X001T007

XOT: SENT TO ROUTER

13:48:15.53

E8E2E7C3 C6E3C1E2 E8E2D7C1 E24040

13:48:15.54

\*

... PESIT SYSXCFTASYSPAS \* 0989117C

00000 00000003 100121 \* .... \* 0989117C X001T007 XOT: SENT TO ROUTER 13:48:15.55

00000 00000007 100120C1 C3D2F0 \* ....ACK0 \* 098A4176 X001T007 XOT: RECEIVED FROM ROUTER 13:48:15.57

00000 00000003 100121 \* .... \* 0989117C X001T007 XOT: RECEIVED FROM ROUTER

00000

00020

00040

00060

00080

00000083 10013200 A0402000 D9030853

42050653 59535041 53060102 07030024

2C443D32 30303530 31303531 33343831

31352D2D 4D565332 3230432D 41323330

3D514334 443248

13:48:15.72

59535843 46544104 08535953 58434654 \*........to ....SYSXCFTA..SYSXCFT\*

02160102 17010163 6E434654 20593D4D \*B..SYSPAS......$.......cnCFT Y=M\*

3536302C 563D3233 302C5A3D 702D312D \*,D=2005010513481560,V=230,Z=p-1-\*

3033352D 32303031 2F31302F 32322C4B \*15--MVS220C-A230035-2001/10/22,K\*

\*=QC4D2H

\*

0989117C

0989119C

098911BC

098911DC

098911FC

X001T007 XOT: SENT TO ROUTER 13:48:15.72

00000 00000003 100141 \* .... \* 0989117C X001T007 XOT: RECEIVED FROM ROUTER

00000 00000023 1001245A 56444850 37444C4E

00020 33333430 333832

13:48:15.72

39374A49 36513153 49594C2C 433D3830 \*...£..$ZVDHP7DLN97JI6Q1SIYL,C=80\*

\*3340382

\*

0989117C

0989119C

X001T007 XOT: SENT TO ROUTER 13:48:15.73

00000 00000003 100161 \* .../ \* 0989117C

*Example of line trace (XOT line)*

X001T007 005: INBOUND CALL PACKET 15:10:11.97

00000 0BF00806 0018800A 42070743 030302CC 0300C018 80105043 4E4531 \*........B..C..........PCNE1 \* 09896176 X001T007 XOT: OUTBOUND X25 COMMAND 15:10:11.99

00000 0F \*. \* 0989617E

X001T007 XOT: INBOUND DATA 15:10:12.08

00000 00C3C6E3 D7E2C9E3 E7C3D7C1 E7F14040 40D7D8D9 E2404040 400D25 \* CFTPSITXCPAX1 PQRS .. \* 09891182 P001I001 AP80LU51 I09: DATA TO CFTBACB1 15:10:12.21

00000 C3C6E3D7 E2C9E3E7 C3D7C1E7 F1404040 D7D8D9E2 40404040 0D25 \*CFTPSITXCPAX1 PQRS .. \* 0989617D P001I001 AP80LU51 I09: DATA FROM CFTBACB1 15:10:12.23

00000 C1C3D2F0 0D25 \*ACK0.. \* 0989617C X001T007 XOT: OUTBOUND DATA 15:10:12.23

00000 00C1C3D2 F00D25 \* ACK0.. \* 098A417C X001T007 XOT: INBOUND DATA 15:10:12.46

00000 22003240 2000D903 0D4F5020 20202020 20435041 5831040D 4F502020 20202020 \*".to ....OP CPAX1..OP \* 09891182

00020 43504258 31060101 07030024 02160100 170101 \*CPBX1......$....... \* 098911A2 P001I001 AP80LU51 I09: DATA TO CFTBACB1 15:10:12.47

00000 00324020 00D9030D 4F502020 20202020 43504158 31040D4F 50202020 20202043 \*.to ....OP CPAX1..OP C\* 098A417D

00020 50425831 06010107 03002402 16010017 0101 \*PBX1......$....... \* 098A419D P001I001 AP80LU51 I09: DATA FROM CFTBACB1 15:10:12.55

00000 00114021 D9E20601 01070300 24021701 01 \* . .RS..... ..... \* 0989617C X001T007 XOT: OUTBOUND DATA 15:10:12.55

00000 00001140 21D9E206 01010703 00240217 0101 \* . .RS..... ..... \* 098A417C X001T007 XOT: INBOUND DATA 15:10:12.89

00000 440028C0 1FE20009 0C0B02FF FF0C0650 434E4531 410D0307 DC1D5C0D 414E5449 \*D.(............PCNE1A.....ç.ANTI\* 09891182

00020 50434E45 20544553 54 \*PCNE TEST \* 098911A2 P001I001 AP80LU51 I09: DATA TO CFTBACB1 15:10:12.90

00000 0028C01F E200090C 0B02FFFF 0C065043 4E453141 0D0307DC 1D5C0D41 4E544950 \*.(............PCNE1A.....ç.ANTIP\* 098A417D

00020 434E4520 54455354 \*CNE TEST \* 098A419D P001I001 AP80LU51 I09: DATA FROM CFTBACB1 15:10:12.97

00000 000BC03F D9000203 000000 \* .é.R .. \* 0989617C X001T007 XOT: OUTBOUND DATA 15:10:12.97

00000 00000BC0 3FD90002 03000000 \* .é.R .. \* 098A417C X001T007 XOT: INBOUND DATA 15:10:33.11

00000 66000B40 23E2D902 03000000 \*. . .SR.. \* 09891182 P001I001 AP80LU51 I09: DATA TO CFTBACB1 15:10:33.12

00000 000B4023 E2D90203 000000 \* . .SR.. \* 098A417D P001I001 AP80LU51 I09: DATA FROM CFTBACB1 15:10:33.14

00000 00064024 D9E2 \* . .RS \* 0989617C X001T007 XOT: OUTBOUND DATA 15:10:33.14

00000 00000640 24D9E2 \* . .RS \* 098A417C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X001T007 | XOT: INBOUND X25 COMMAND | 15:10:33.27 |  | |
| 00000 130000 |  |  | \*. | \* 09891182 |
| X001T007 | XOT: OUTBOUND X25 COMMAND | 15:10:33.27 |  |  |
| 00000 17 |  |  | \*. | \* 09891182 |

*Example of “trace by rule” (XOT terminal to application on /PCNE line)*

**6.6. VIRTEL SNAP**

VIRTEL maintains an internal trace table in which it records significant events which occur during VIRTEL processing. The SNAP command allows the administrator to obtain a snapshot listing of the contents of the trace table at a given point in time.

The SNAP listing is primarily intended for use by VIRTEL development personnel and will normally need to be forwarded to Syspertec for analysis. For customer diagnostics, the TRACE command (described above) may often be more useful.

The format, the contents, and the size of the SNAP depend on the SNAPW, TRACBIG, TRACEB, TRACEON parameters in the VIRTCT.

The internal trace table is recorded in a circular fashion, so that each new event added to the table overlays and replaces the oldest event in the table. The table contains a fixed number of event slots (determined by the TRACBIG parameter in the VIRTCT), and additionally certain events may have a variable amount (up to 256 bytes) of data recorded. The variable data is stored in a separate area whose size is determined by the TRACEB parameter in the VIRTCT, and this area is also filled in a circular manner, with the oldest information being dropped from the table when new information is added. Thus, depending on the values of the TRACBIG and TRACB parameters, older entries in the trace table may no longer have data associated with them.

Clearly, the greater the level of activity in the VIRTEL system, the quicker the trace table will wrap and information will be pushed out to make way for new entries. Thus, in order for the SNAP listing to provide useful information, the size of the trace table and its associated buffer area must be adequate for the level of system activity, and the SNAP command must be issued as quickly as possible after the event under investigation occurs. In some cases it may be necessary to use an automation tool to issue the SNAP command immediately following the appearance of a certain console message.

As well as dumping the contents of the internal trace table, the SNAP command also dumps certain VIRTEL internal control blocks. A dump of the control blocks associated with a particular terminal may optionally be requested.

The SNAP command is described under the heading [“VIRTEL commands”, page 1](#_bookmark0) and [“Obtaining a SNAP”, page 19](#_bookmark56). VIRTEL may also produce a SNAP listing automatically if a program check or other abend occurs during VIRTEL processing.

In **MVS environment**, the SNAP output is written to the SYSPRINT file in the VIRTEL started task. In **VSE environment**, the SNAP output is written to the POWER LST file of the VIRTEL partition.

Several SNAP commands may be issued during a single run of VIRTEL. The output file may thus contain successive SNAP listings concatenated one after the other.

P S W

00000 00000000 8002E290 PSW = VIR0009 +2BB8

R E G I S T E R S

00000 00000000 07F48830 07F48838 0001132C

00020 07F474A8 080495C0 000281D8 000CEA10 R10 = VIR0099 +0000

R12 = VIR0009 +2A60 R14 = VIR0009 +2BB8 R15 = VIR0009 +2412

DATE=18 Oct 2005 TIME= 17:22:26 TASK= SYSPERTEC COMMUNICATION

196 BUREAUX DE LA COLLINE 92213 SAINT CLOUD CEDEX P500 - PERMANENT

0003E978

\*

..S.

\*

0003E980

07F488C0 00000100 0804A2CB 00000410

0002E138 07F48758 8002E290 0002DAEA

\*

.4h0.4h8 ..,.4hé . ..s. ..\*

\*.4ty..né .aQ ... ..8.4gX..S. ...\*

ABEND=SNAP TERM=NTIN0000 APPLICATION=SP3VIRMB VERSION=4.32

TERMINAL TASK R15

R14

MODULE

OFFS FUNCTION

1/10000 S.

.............

**6.6.1. Example of SNAP**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| NTOUT009 | 0253 | 00000000 | 8002DFD6 | VIR0009 +28FE | 3032 | CONTINUE | 1640023458 |
| NTOUT009 | 0253 | 08043B80 | 8002E082 | VIR0009 +29AA | 8042 | $FREEMAIN | 1640023458 |
| NTOUT009 | 0253 | 00000000 | 8002E082 | VIR0009 +29AA | 8021 | $RECANY | 1640023458 |
| NTOUT009 | 0253 | 07F472D8 | 8002EDDC | VIR0009 +3704 | 8042 | $FREEMAIN | 1640023458 |
| NTTCP-LI | EE54 | 421E0006 | 800565AC | VIR0T19 +02AC | 804E | $CREATE | 1640023459 |
| NTTCP-LI | 0254 | 07F580B0 | 8002FF58 | VIR0009 +4880 | 8040 | $GETMAIN | 1640023459 |
| NTTCP-LI | 0254 | 00000005 | 8003008E | VIR0009 +49B6 | 804D | TO LINE | 1640023459 |
| NTTCP-LI | 0254 | 00000000 | 80032146 | VIR0M13 +00A6 | 8043 | TO TCP | 1640023459 |

|  |  |  |  |
| --- | --- | --- | --- |
| 00000 | 07F49108 0400011C F0F0F0F0 F0F0F0F0 | 00000000 00076108 00076108 07F18D60 | \*.4j.. ..00000000 .a. .a..1.-\* |
| 00020 | FC040000 00000000 00000000 00000000 | 00000000 00000000 00000000 00000000 | \*.. \* |
| 00040 | 00000000 00000000 00000000 00000000 | 00000000 00000100 00000000 00000000 | \* . \* |
| 00060 | 00000000 00000000 00000000 00000000 | 00000000 00000000 00000000 00001200 | \* \* |

*Example of SNAP listing*

A C T I V E R E Q U E S T 00000

00020

00040

00060

00080

00000000 02000160 D9C5C1C4 40404040

00000000 00000000 B4CFD8A6 95840500

07F579A0 080495C8 00000000 00000000

07F575A8 00072EEC 07F46790 08049868

00000000 00000000 00000000 00000000

... SAME AS ABOVE ...

00000000 00000000 00000000 00000000

07F575A8 00000000 07F5D688 00071A80

00038D00 00000000 00000000 00000000

00000000 00000000 00000000 00000000

07F57560

00030000 00000000 00000000 00000000

07F18F6C 800761AC 07F18D60 07F46780

00000000 00000000 07F18FA4 07F575AC

0804A2D0 00000000 87F57658 00000000

00000000 00000000 00000000 00000000

\*

\*

. .-READ

.

\*

\*.5y...nH

..Qwnd. .1.l..a..1.-.4g.\*

.1.u.5u.\*

\*.5uy ....4g...qh..sè

\*

g5vX

\*

\*

000E0

00100

00120

00140

00000000 00000000 07F57560 07F575AC

07F18D60 07F57560 00000000 7F69E1D0

00000000 00000000 00000000 00000000

00000000 00000000 00000000 00000000

07F49108

\*

\*.5uy

\* ..

\*

.5u-.5u.\*

.5Oh ....1.-.5u- .i.è\*

\*

\*

S E S S I O N

**6.6.2. Message-triggered SNAPMSG**

The SNAPMSG command requests VIRTEL to generate an automatic SNAP after certain messages (VIRI902W, VIR0026W, VIR0052I, VIR1552I, VIR0526W or VIR1952I).

Only one SNAP can also be obtained with user specific code provided by SYSPERTEC for messages VIRHT31E and VIRHT63E. See [“VIRTEL commands”, page 1](#_bookmark0)

# Statistics

**7.1. The VIRSTAT File**

The VIRSTAT file is a sequential file into which VIRTEL writes connection statistics.

When the STATS=YES parameter is coded in the VIRTCT, the VIRSTAT file is reinitialised at each VIRTEL startup. With STATS=YES, VIRTEL must be stopped periodically in order to avoid filling the VIRSTAT file, and the file should be defined as a GDG in order not to lose the information from a previous run. The STATS=MULTI parameter may be coded in the MVS environment to permit continuous operation.

Trying to browse the contents of the file from ISPF while it is in use by VIRTEL, can result to the obtain a system message indicating that the file is empty. In reality this is not true, because in fact the records are buffered in memory before being written in block.

The format of the statistics records depends on the value specified in the “Write Stats to” field of the VIRTEL terminal definition (refer to the VIRTEL Connectivity Reference manual for details of terminal definitions). Each terminal may request statistics in one or more of the possible formats:

### Classic

VIRSTAT classic format recording is intended for use with Minitel calls on terminals associated with NPSI lines (Gate or Fast Connect).

### Alternate X25

VIRSTAT alternate format recording may be requested for terminals associated with any X25 line (GATE, FASTC, XOT).

### Web

VIRSTAT format suitable for terminals associated with an HTTP line.

For terminals associated with all other line types (including /GATE, /PCNE, and /FASTC) the statistics record may not contain meaningful information and the statistics field in the terminal definition should be left blank.

The statistics file may contain a mixture of classic, alternate X25, and web format records. The record type indicator at position 61 of each record identifies the format of the particular record.

**7.1.1. VIRSTAT classic format**

For terminals which specify classic format recording (STATS=1), the VIRSTAT record format is shown in the following table:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

*Format of VIRSTAT record (classic format)*

For NPSI Fast Connect lines, the X25 accounting statistics are recorded in billing units provided by NPSI at virtual circuit disconnection time. Their interpretation depends on the TAXUNIT parameter in the NPSI X25.MCH macro. Similarly, the session start and end times (Fast Connect only) are provided by NPSI and depend on the clock settings in the NCP. For other types of lines, accounting statistics and times are generated by VIRTEL.

### Note 1

Type C (cumulative) records are implemented at terminal disconnection. Type P (partial) records are implemented at regular intervals.

Type E (end of job) records are implemented at VIRTEL shutdown.

**7.1.2. VIRSTAT alternate X25 format**

For terminals which specify alternate X25 format recording (STATS=4), the VIRSTAT record format is shown in the following table:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

*Format of VIRSTAT record (alternate X25 format)*

For NPSI Fast Connect lines, the X25 session start and end times are provided by NPSI at virtual circuit disconnection time and depend on the clock settings in the NCP. For other types of lines, times are generated by VIRTEL.

### Note 1

Type I (inbound) records relate to X25 incoming calls. Type O (outbound) records relate to X25 outgoing calls.

**7.1.3. VIRSTAT formats for VIRTEL Web Access**

For terminals which specify web format recording (STATS=5 or STATS=6), the VIRSTAT record format is shown in the following tables:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

*Format of VIRSTAT record (type 5 for Web Access)*

This record type is written when 5 is specified in the STATS field of the terminal definition used for the HTTP line. If the terminal is disconnected by TIMEOUT, the “Error Code” field contains the word “TIME”.

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

*Format of VIRSTAT record (type 6 for Web Access)*

For this record type, the counters are in binary, and the Session Date and Time fields are replaced by User name (20 bytes) and URL parameter (first 16 bytes).

This record type is written when 6 is specified in the STATS field of the terminal definition used for the HTTP line.

**7.1.4. Statistics file management**

The STAT command is used to manage the VIRTEL statistics recording files (VIRSTATx). This command can be used only if STATS=MULTI is specified in the VIRTCT.

**7.1.4.1. Display VIRSTAT**

STAT,D

This command displays the status of the VIRSTATx files (message VIR0601I).

**7.1.4.2. Switch VIRSTAT**

STAT,I

This command forces VIRTEL to free the current VIRSTATx file and to start recording onto the next file.

**7.1.5. Printing the contents of the VIRSTAT file (X25)**

The VIR0070 program allows the contents of the VIRSTAT file to be printed. The source for this program is supplied in the SSL (VSE) or in the SAMPLIB (MVS) and you can use this as the basis of a user-written program to print statistics. Examples of the JCL required to execute this program are shown below:

* $$ JOB JNM=VIRSTAT,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=V,DEST=(,SPTUSER)

// JOB VIRPRNT

// LIBDEF \*,SEARCH=VIRT442.SUBLIB

// DLBL STAT,'VIRTEL.VIRSTAT.ESDS',,VSAM,CAT=VSESPUC

// EXEC VIR0070,SIZE=AUTO

/\*

/&

\* $$ EOJ

*VIR0070 JCL to print VIRSTAT file (VSE)*

//VIRSTAT JOB

//TRI

EXEC

//SYSPRINT DD

//SYSOUT

DD

//SORTWK01 DD

//SORTWK02 DD

//SORTIN

//SORTOUT

//

//

//SYSIN

DD

DD

DD

1,USER,CLASS=A,MSGCLASS=X,NOTIFY=&SYSUID PGM=SORT

SYSOUT=\* SYSOUT=\*

UNIT=3380,SPACE=(TRK,(100,20),RLSE) UNIT=3380,SPACE=(TRK,(100,20),RLSE) DISP=SHR,DSN=VIRTEL.STAT DSN=&&STAT,UNIT=SYSDA,DISP=(,PASS), DCB=(LRECL=124,BLKSIZE=620,RECFM=FB), SPACE=(TRK,(100,20),RLSE)

\*

SORT FIELDS=(1,16,A),FORMAT=CH END

//STAT EXEC PGM=VIR0070

//STEPLIB DD DISP=SHR,DSN=VIRT442.LOADLIB

//SYSUDUMP DD SYSOUT=\*

//SYSPRINT DD SYSOUT=\*

//VIRSTAT DD DISP=(OLD,DELETE),DSN=&&STAT

//

*VIR0070 JCL to print VIRSTAT file (MVS)*

**7.1.6. Printing the contents of the VIRSTAT file (Web)**

The PRTSTATW program supplied with the system allows printing of type 6 records from the VIRSTAT file. This program is delivered as a load module in the VIRTEL LOADLIB (from version 4.45 onwards) and the execution JCL is provided as member JCLPRTST in the VIRTEL SAMPLIB. Examples of the execution JCL for this program are shown below.

**7.1.6.1. PRTSTATW JCL**

In the VSE environment the VIRPRTST job, loaded into the POWER reader queue during VIRTEL installation, contains an example of JCL for printing the VIRSTAT file. This job is an example only and must be modified before execution:

* + $$ JOB JNM=VIRPRTST,CLASS=0,DISP=D
  + $$ LST DA

// JOB VIRPRTST

\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* + \* VIRTEL: EXAMPLE JCL TO EXECUTE STAT VIRTEL PRINT \*

\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// LIBDEF \*,SEARCH=(VIRT452.SUBLIB,PRD2.CONFIG,PRD1.BASE)

// DLBL SORTIN1,'VIRTEL.STAT',,VSAM,CAT=VSESPUC

// DLBL SORTOUT,'%VIRTEL.SORTFILE',0,VSAM,CAT=VSESPUC,DISP=(NEW,KEEP), C RECORDS=(10,100),RECSIZE=124

* // DLBL SORTOUT,'VIRTEL.SORTFILE',0,SD
* // EXTENT SYS001,SYSWK2,1,0,NNNN,15
* // ASSGN SYS001,DISK,VOL=SYSWK2,SHR

// EXEC SORT,SIZE=100K

SORT FIELDS=(01,08,A),FORMAT=CH RECORD TYPE=F,LENGTH=124

END

/\*

* OPTIONS FOR PRINT OR COUNT
* // DLBL SYSPRINT DD SYSOUT=\*
* // DLBL SYSABEND DD SYSOUT=\*
* // DLBL IJSYSLS DD SYSOUT=\*

\*

// DLBL VIRSTAT,'%VIRTEL.SORTFILE',0,VSAM,CAT=VSESPUC,DISP=(,DELETE)

// ASSGN SYS007,SYSLST

// ASSGN SYS006,SYSRDR

// EXEC PGM=PRTSTATW,SIZE=AUTO

PRTSTATW PRINT NNN 0250

SELDATE 01012011 31122012 SELTERM DEVTA\* CLVTA\*

/\*

* \* MAIN CARD (REQUIRED)

\* \* ---------

\* \* 1 2 3 4 5 6

\* \*1...!....0....!....0....!....0....!....0....!....0....!....0...

* \*PRTSTATW PRINT BREAK USER NNN PPPP
* \* SPECIFIC REQUEST TO PRINT THE STATISTICS
* \*PRTSTATW COUNT $ALL$ NNN PPPP
* \* SPECIFIC REQUEST TO COUNT THE NUMBERS OF DIFFERENT
* \* USERS
* \*SELECT CARD (OPTIONAL)

\* \* -----------

\* \* 1 2 3 4 5 6

\* \*1...5....0....5....0....5....0....5....0....5....0....5....0

* \*SELDATE DDMMYYYY DDMMYYYY
* \* DATE SELECTION BEGIN,
* \* END DATE
* \*SELTERM TTTTTTTT XXXXXXXX YYYYYYYY ZZZZZZZZ (UP TO 8 BYTES)

\* \*SELUSER USER4561890123456789 (UP TO 20 BYTES)

* \*SELPARM PARM456189012345 (UP TO 16 BYTES)

\* \*1...!....0....!....0....!....0....!....0....!....0....!....0...

* \* THE '\*' CHARACTER ALLOWS A GENERIC EVALUATION.

/&

* $$ EOJ

*PRTSTATW JCL to print VIRSTAT file in VSE (type=6)*

In the MVS environment the JCL for executing the PRTSTATW program is supplied as member JCLPRTST in the VIRTEL SAMPLIB:

//VIRPRTST JOB 1,USER,CLASS=A,MSGCLASS=X,NOTIFY=&SYSUID

/\*JOBPARM LINES=9999

//\*------------------------------------------------------------\*

//\* VIRSTAT FILE PRINTING JOB \*

//\* VIRSTAT RECORD TYPE 6 (VSTA-RECORD-TYPE='B') \*

//\*------------------------------------------------------------\*

// SET LOAD=yourqual.VIRTnnn.LOADLIB

// SET STAT=yourqual.VIRTnnn.STAT

//\*------------------------------------------------------------\*

//JOBLIB DD DISP=SHR,DSN=&LOAD

//\*------------------------------------------------------------\*

//\*

//STEP0 EXEC PGM=SORT

//SYSOUT DD SYSOUT=\*

//SORTIN DD DISP=SHR,DSN=&STAT

//SORTOUT DD DSN=&&SORTSTAT,DISP=(NEW,PASS),UNIT=SYSDA,

// DCB=(BLKSIZE=1240,LRECL=124,RECFM=FB),

// SPACE=(TRK,(1,1))

//SYSIN DD \*

SORT FIELDS=(1,8,A),FORMAT=CH

//\*

//STEP1 EXEC PGM=PRTSTATW

//SYSPRINT DD SYSOUT=\*

//SYSABEND DD SYSOUT=\*

//IJSYSLS DD SYSOUT=\*

//\*

//VIRSTAT DD DSN=&&SORTSTAT,DISP=OLD

//SYSIN DD \*

PRTSTATW PRINT NNN 0250

SELDATE 01012011 31122012 SELTERM DEVTA\* CLVTA\*

\* ..!....0....!....0....!....0....!....0....!....0....!....0...

//\* MAIN CARD (required)

//\* ---------

//\* 1 2 3 4 5 6

//\* 1...!....0....!....0....!....0....!....0....!....0....!....0...

//\* PRTSTATW PRINT BREAK USER NNN PPPP

//\* specific request to print the statistics

//\* PRTSTATW COUNT $ALL$ NNN PPPP

//\* specific request to count the numbers of different

//\* users

//\* SELECT CARD (optional)

//\* -----------

//\* 1 2 3 4 5 6

//\* 1...5....0....5....0....5....0....5....0....5....0....5....0

//\* SELDATE DDMMYYYY DDMMYYYY

//\* DATE selection begin,

//\* end date

//\* SELTERM TTTTTTTT XXXXXXXX YYYYYYYY ZZZZZZZZ (up to 8 bytes)

//\* SELUSER USER4561890123456789 (up to 20 bytes)

//\* SELPARM PARM456189012345 (up to 16 bytes)

//\* 1...!....0....!....0....!....0....!....0....!....0....!....0...

//\* The '\*' character allows a generic evaluation.

//

*PRTSTATW JCL to print VIRSTAT file in MVS (type=6)*

This JCL consists of two main steps:

* a first step to sort the file
* a second step to PRINT or COUNT the records

**7.1.6.2. Sorting the file**

The sort requirements are determined by the type of report desired. Since the PRTSTATW program offers the option of selecting records and also offers up to two levels of report break to allow printing of subtotals, it is important to specify the appropriate sort criteria to obtain the correct result.

The sort operates on one or more criteria, in ascending (A) or descending (D) mode. You should adapt the SORT SYSIN according to the syntax of the specific SORT program being used.

Several examples of sort criteria are shown below for various fields: terminal (TERM), date (DATE), user name (USER), URL parameter (PARM)

SORT FIELDS=(1,8,A) SORT FIELDS=(9,4,A) SORT FIELDS=(89,20,A)

SORT FIELDS=(109,16,A)

--> TERM

--> DATE

--> USER

--> PARM

A: ascending D: descending

SORT FIELDS=(1,8,A,),FORMAT=CH sort by TERM

SORT FIELDS=(1,8,A,89,20,A),FORMAT=CH sort by TERM first then USER

SORT FIELDS=(17,15,A) --> IP Adress (for $ALL$ request)

SORT FIELDS=(89,20,A,17,15,A),FORMAT=CH sort for $ALL$ request

*PRTSTATW JCL sort criteria*

For example, to obtain a report in ascending order of session start date, specify the following statements in the SORT SYSIN:

//SYSIN

DD \*

SORT FIELDS=(9,4,A),FORMAT=CH

//\*

**7.1.6.3. The PRTSTATW program**

The PRTSTATW program executed in the second step reads the sorted output file from the first step. It contains required and optional SYSIN cards.

**7.1.6.3.1. First card (required)**

1

2

3

4

5

6

1...!....0....!....0....!....0....!....0....!....0....!....0...

PRTSTATW PRINT BREAK USER NNN PPPP

*PRTSTATW first SYSIN card*

### Columns 1 to 8

Program name: must be PRTSTATW

### Columns 11 to 16

Report type: specify PRINT (print report) or COUNT (calculate number of distinct users)

### Columns 31 to 35

Optionally specify BREAK if report break is desired (up to 2 levels) for printing (PRINT) or $ALL$ if counting (COUNT)

### Columns 37 to 40

Optionally indicates the type of report break: TERM (break on change of terminal name), USER (break on change of user name), DATE (break on change of date) or PARM (break on change of URL parameter)

### Columns 43 to 46

Optionally indicates the second level report break (TERM, USER, DATE, or PARM)

### Columns 51 to 53

Optionally specify N (no) or O (yes) to print additional trace information (program trace, input/output trace, and miscellaneous trace respectively). The default is N for each trace.

### Columns 56 to 59

Maximum number of pages to be printed (default 50 pages)

**7.1.6.3.2. Second card (optional)**

1

2

3

4

5

6

1...5....0....5....0....5....0....5....0....5....0....5....0

SELDATE

SELTERM SELUSER SELPARM

DDMMYYYY DDMMYYYY

DATE selection begin,

end date

TTTTTTTT XXXXXXXX YYYYYYYY ZZZZZZZZ (up to 8 bytes) USER4561890123456789 (up to 20 bytes)

PARM456189012345

(up to 16 bytes)

*PRTSTATW second SYSIN card*

This card allows records to be selected according to 4 fields: DATE (selection by date range), TERM (selection of up to 4 different terminal names, otherwise 4 different HTTP lines), USER (selection by user name), PARM (selection by URL parameter).

It is possible to make a **generic** selection by coding a ‘\*’ character at the end of a field. For example, specifying a terminal selection value of DEVT\* allows the program to select all records whose terminal name begins with DEVT.

### Columns 1 to 7

Optional, indicates the selection type: SELDATE (for DATE), SELTERM (for terminal), SELUSER (for user), or SELPARM (for URL parameter).

### Columns 11 to 19

Indicates up to 8 characters for the chosen value (SELDATE and SELTERM). The value may end in ‘\*’ for a generic search.

### Columns 21 to 29

For SELDATE: second date in the range, for SELTERM: second terminal name (optional)

### Columns 31 to 39

For SELTERM: third terminal name (optional)

### Columns 41 to 49

For SELTERM: fourth terminal name (optional)

### Columns 21 to 40

For SELUSER:up to 20 characters for the user name. The value may end in ‘\*’ for a generic search.

### Columns 21 to 36

For SELPARM: up to 16 characters for the URL parameter. The value may end in ‘\*’ for a generic search.

**7.1.6.4. Counter report**

Clients who wish to obtain the total number of unique users can execute the PRTSTATW program with the SYSIN shown below.

For the SORT: the first sort field is the user name, and the second sort field is the IP address:

//SYSIN

DD \*

SORT FIELDS=(89,20,A,17,15,A),FORMAT=CH

//\*

For the PRTSTATW program:

//SYSIN DD \*

PRTSTATW COUNT

$ALL$

NNN 0465

SELDATE

SELTERM

01012011 30122011

DEVTA\*

CLVTA\*

COUNT and $ALL$ are required. The selection cards are optional. They allow for example to report for a given period the number of different users connected to the system in HTTP mode and/or to filter on an HTTP line defined with transaction security active, which requires the user to sign on.

For MVS, sample JCL for the user counter report is supplied in the JCLCOUST member of the VIRTEL SAMPLIB. For VSE, a sample job named VIRCOUST is loaded into the POWER Reader Queue at installation time.

**7.1.6.4.1. Example of counter report (COUNT)**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* TOP OF DATA

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1== VIRTEL == Statistics file COUNT job ==

Submit on: 18-01-2011 at: 14:23:12 PAGE : 000

Parameters CARDS list read by PRTSTATW

PRTSTATW COUNT

$ALL$ PARM

NNN 0469

SELDATE 01012010 30122011

\* ..!....0....!....0....!....0....!....0....!....0....!....0...

18-01-2011 14:23:12

18-01-2011 14:23:12

at top of listing:

18-01-2011 14:23:12

\* S U M M A R Y \*

With criterias put for selection

Total Records read

:

00207

18-01-2011 14:23:12 Total Calls selected

: 00017

18-01-2011 14:23:12

Total Calls duration

: 001hr04mn35s

18-01-2011 14:23:12

Total Calls ended by "Timeout"

: 00007

18-01-2011 14:23:12

Total Defined different Users

: 00004

18-01-2011 14:23:12

Total Calls without signature

: 00001

PRTSTATW 18-01-2011 14:23:12 End of execution

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* BOTTOM OF DATA \*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

*PRTSTATW user counter report*

**7.1.7. SMF Support**

Using VIRTEL 4.53+ and onwards allows VIRTEL SMF support writing VIRSTATS records into SMF. The VIRTCT must be reassembled and link-edited with a new value SMF or (SMF,nnn) for the STATS parameter to have this feature active.

The SMF record format is the same as the current STATS record but prefixed by the standard SMF header. The default SMF record number is 223, but it can be modified using the (SMF,nnn) syntax.

**7.1.8. Printing the contents of the VIRSTAT SMF records**

The SMFPRINT job in VIRTEL.SAMPLIB can be used to print the SMF records from the SYS1.MANx dataset using SMFREXXP REXX procedure.

**7.1.9. Messages**

Messages "VIR0612E VIRSTAT SMFWTM FAILED. RC=rc" and "VIR0611I VIRSTAT NOW RECORDING TO SMF" are in

relation with SMF support. See "Virtel Messages and Operations" manual for more details.

# Memory management

The VIRTEL memory management sub-application allows the system administrator to display VIRTEL memory utilisation in real time. The memory management sub-application is a pseudo-graphical display which shows the allocation of VIRTEL memory by function. VIRTEL manages its own memory, in order to avoid memory shortages as a result of fragmentation. The memory management display can be used by the administrator to help understand VIRTEL’s memory requirements during normal operation.

**8.1. Access To The Application**

To invoke the memory management sub-application, press [PA2] in the Configuration Menu to display the Sub- Application Menu, then press [PF4] in the Sub-Application Menu. The sub-application displays a screen similar to the example shown below. This screen represents the contents of the VIRTEL address space after deducting the space occupied by the VIRTEL kernel modules.

|  |  |  |  |
| --- | --- | --- | --- |
| .=Free block 4=EIB-Session | 1=Permanent 5=Tioa-Work | 2=Temporary  6=Sub-application | 3=Screen |
| P3=Return | P6=1st Page | P7=Page-1 | P8=Page+1 |

*Memory display of VIRTEL address space*

MEMORY BLOCKS USAGE -------------------------------- Applid: SPVIRD2 10:49:45

System :

1046 K Data

:

5824 K Maximum :

6870 K

00000000 .............1111111............................................

07C00000 ...................................................6566666664235

08000000 5555553553333533335553555355535553555355535553555355553555355535

08400000 5555355553555535335555355553555535555355553555535555355553555355

08800000 553555355333353333..............................................

Each screen position represents a 2K memory block (if MEMORY=BELOW is specified in the VIRTCT), or a 64K memory block (if MEMORY=ABOVE). The address displayed at the start of each line is the virtual address represented by the first position in the line.

Each free memory block is represented by a dot. Lines which consist entirely of dots are not displayed.

Permanently allocated memory blocks are represented by the character 1. To avoid memory shortages as a result of fragmentation, these blocks are always allocated at the end of the VIRTEL address space.

Temporarily allocated memory blocks are represented by the character 2. Blocks of this type are allocated and freed by VIRTEL as required.

Memory blocks used by the VIRTEL Multi-Session feature to save screen images are represented by the character 3. Blocks of this type are allocated and freed by VIRTEL as required.

Memory blocks used for saving EIB and other session-related information are represented by the character 4. Blocks of this type are allocated and freed by VIRTEL as required.

Memory blocks used as communication areas by VIRTEL sub-applications are represented by the character 5. Blocks of this type are allocated and freed by VIRTEL as required.

In the VSE environment, sub-application modules are loaded in the SUBPOOL. Memory blocks used for this purpose are represented by the character 6. Blocks of this type are allocated and freed by VIRTEL as required. These blocks never appear in the MVS environment.

**8.1.1. Memory display in MEMORY=TEST mode**

If MEMORY=TEST is specified in the VIRTCT, the memory management sub-application displays its results in a different format. MEMORY=TEST mode allows support technicians to analyse memory occupation by module, as a debugging aid for possible memory shortage problems.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | System | : 1046 K Data : 5824 K Maximum : 6870 K | | |
| 0004E208 | VIR0009 | +26A8 | | |
| 0000E2B3 | VIR0009 | +4A4A | | |
| 00004B91 | VIR0009 | +2E60 | | |
| 00004B48 | VIR0000 | +5C8C | | |
| 000026E2 | VIR0000 | +5DF6 | | |
| 00001110 | VIR0009 | +4BC2 | | |
| 00000F22 | VIR0000 | +25B6 | | |
| 00000C12 | VIR0000 | +5F86 | | |
| 000009D8 | VIR0000 | +6280 | | |
| 000009C0 | VIR0000 | +3226 | | |
| 000008E8 | VIR0T09 | +2270 | | |
| 000007CC | VIR0000 | +6138 | | |
| 00000524 | VIR0B17 | +47D8 | | |
| 00000420 | VIR0T09 | +10D2 | | |
| 00000378 | VIR0T09 | +2962 | | |
| 00000270 | VIR0I09 | +056E | | |
| .=Free block | | 1=Permanent | 2=Temporary | 3=Screen |
| 4=EIB-Session  P3=Return | | 5=Tioa-Work  P6=1st Page | 6=Sub-application  P7=Page-1 | P8=Page+1 |

*Memory display in MEMORY=TEST mode*

MEMORY BLOCKS USAGE -------------------------------- Applid: SPVIRD2 13:17:23

Each line of the screen represents one VIRTEL module which has obtained one or more memory blocks. The first column represents the number of bytes of memory (en hexadecimal) currently allocated by the module. The first 16 modules are displayed, in descending order of memory utilisation.

**8.1.2. Positioning the display**

Where the memory display occupies more than one screen, you can press [PF8] to view the following page, [PF7] to view the previous page, and [PF6] to go back to the first page.

**8.1.3. Real time monitoring**

To refresh the display with up-to-date information, press [Enter].

**8.1.4. Return to the sub-application menu**

To return to the sub-application menu, press [PF3] or [Clear]

**8.2. Virtel Memory Display**

The Memory display feature is a memory diagnostic tool created to trap possible invalid Virtel memory free requests. Such request can lead to ABEND0C4s and other unwanted behaviour. Virtel memory requests (PRENDRE and RENDRE) are tracked in a diagnostic storage area located above the bar.

The area is 1MB in size and can contain 65536 active storage requests. An active storage request is a storage area that has been gotten (PRENDRE) and is pending a Virtel storage release (RENDRE).

This diagnostic tool should only be used when recommended by Technical Support.

# Memory trace

# management

A memory trace can be activated using a command or from the VIRTCT. In both case, VIRTEL records an history of memory allocations that appears in a SNAP listing.

**9.1. Memory Trace Commands**

**9.1.1. Activating memory trace**

A memory trace can be activated by using

MEMTRACE

The first answer is in the form:

VIR0200I MEMTRACE

VIR0214I MEMORY TRACE STARTED

VIR0218I MEMORY TRACE FOUND 00000000 BLOCKS USING 0000000000000000 BYTES (00000000 MEGS)

The following answers are in the form:

VIR0200I MEMTRACE

VIR0218I MEMORY TRACE FOUND 00000011 BLOCKS USING 0000000000053344 BYTES (00000000 MEGS)

**9.1.2. Reseting memory trace**

A memory trace can be reseted by using

MEMTRACE,CLEAR

The trace is stopped, memory blocks used by the memory trace are released, the trace is restarted.

VIR0200I MEMTRACE,CLEAR

VIR0218I MEMORY TRACE FOUND 00000011 BLOCKS USING 0000000000053344 BYTES (00000000 MEGS) VIR0216I CLEARING MEMORY TRACE

VIR0217I MEMORY TRACE CLEARED VIR0214I MEMORY TRACE STARTED

VIR0218I MEMORY TRACE FOUND 00000000 BLOCKS USING 0000000000000000 BYTES (00000000 MEGS)

**9.1.3. Stopping memory trace**

A memory trace can be stopped by using

NOMEMTRACE

The trace is stopped, memory blocks used by the memory trace are released.

VIR0215I MEMORY TRACE STOPPED

VIR0218I MEMORY TRACE FOUND 00000011 BLOCKS USING 0000000000053344 BYTES (00000000 MEGS)

**9.2. Memory Trace From The VIRTCT**

A memory trace can be activated from the VIRTCT by using MEMORY=TEST or MEMORY=(ABOVE,TRACE) parameter. In such case, the is no message VIR0218I display in the log, but only the benefit of recording the history of memory allocations is kept in the SNAP.

Since it is not possible to stop a trace initialized in this way, it is best to only use this method to perform an analysis of the memory allocation during the startup phase.

**9.3. Memory Trace Analysis**

Once a memory trace activated, issuing a SNAP command produce a report of the memory allocations history in the SNAP listing.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| HTTP-SPV | 0295 | 1EC67F80 | 9EB0E12A | VIR0T09 +206A | 8040 | STILL HERE 1205180941 | 02000208 | 1F4C9220 |
| HTTP-SPV | 0295 | 1EC564E8 | 9EB10BDA | VIR0T09 +4B1A | 8040 | STILL HERE 1205180852 | 04000173 | 1F4C9520 |
| HTTP-SPV | 0295 | 1EC56668 | 8003D63E | VIR0009 +54D6 | 8040 | STILL HERE 1205180792 | 04000270 | 1F088FA0 |
| HTTP-SPV | 0294 | 208A5300 | 8003D458 | VIR0009 +52F0 | 8040 | STILL HERE 1205180792 | 05008CF5 | 1F4C9320 |
| SPVTA015 | 0279 | 209AE880 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180750 | 03000F70 | 1F90A6E0 |
| SPVTA014 | 0279 | 209AF7F8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180705 | 03000F70 | 1F90A660 |
| SPVTA013 | 0279 | 209B0770 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180668 | 03000F70 | 1F90A5E0 |
| SPVTA012 | 0279 | 209B16E8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180630 | 03000F70 | 1F90A560 |
| SPVTA011 | 0279 | 209B2660 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180377 | 03000F70 | 1F90A4E0 |
| SPVTA010 | 0279 | 209B35D8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180318 | 03000F70 | 1F90A460 |
| SPVTA009 | 0279 | 209B4550 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180241 | 03000F70 | 1F90A3E0 |
| SPVTA008 | 0279 | 209B54C8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180149 | 03000F70 | 1F90A360 |
| SPVTA007 | 0279 | 209B6440 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180101 | 03000F70 | 1F90A2E0 |
| SPVTA006 | 0279 | 209B73B8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205180066 | 03000F70 | 1F90A260 |
| SPVTA005 | 0279 | 209B8330 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179956 | 03000F70 | 1F90A1E0 |
| SPVTA004 | 0279 | 209B92A8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179923 | 03000F70 | 1F90A160 |
| SPVTA003 | 0279 | 209BA220 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179881 | 03000F70 | 1F90A0E0 |
| SPVTA002 | 0279 | 209BB198 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179849 | 03000F70 | 1F90A060 |
| SPVTA001 | 0279 | 209BC110 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179782 | 03000F70 | 1F4C9FE0 |
| SPVTA000 | 0279 | 209BD088 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179741 | 03000F70 | 1F4C9F20 |
| SPLOC009 | 0279 | 2099E880 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179715 | 03000F70 | 1F4C9E20 |
| SPLOC008 | 0279 | 2099F7F8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179689 | 03000F70 | 1F4C9D20 |
| SPLOC007 | 0279 | 209A0770 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179654 | 03000F70 | 1F4C9C20 |
| SPLOC006 | 0279 | 209A16E8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179607 | 03000F70 | 1F4C9B20 |
| SPLOC005 | 0279 | 209A2660 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179568 | 03000F70 | 1F4C9A20 |
| SPLOC004 | 0279 | 209A35D8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179526 | 03000F70 | 1F4C9920 |
| SPLOC003 | 0279 | 209A4550 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179488 | 03000F70 | 1F4C9820 |
| SPLOC002 | 0279 | 209A54C8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179458 | 03000F70 | 1F4C9620 |
| SPLOC001 | 0279 | 209A6440 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179377 | 03000F70 | 1F4C9420 |
| SPLOC000 | 0279 | 209A73B8 | 8003A9C0 | VIR0009 +2858 | 8040 | STILL HERE 1205179338 | 03000F70 | 1F4C9120 |
|  | 0000 | 1EC584A8 | 8001BFE4 | VIR0000 +4224 | 8040 | STILL HERE 1205122226 | 04000270 | 1EC48F00 |
|  | 0000 | 1EC58720 | 8001BFE4 | VIR0000 +4224 | 8040 | STILL HERE 1205122225 | 04000270 | 1EC48F20 |
|  | 0000 | 1EC58998 | 8001BFE4 | VIR0000 +4224 | 8040 | STILL HERE 1205122225 | 04000270 | 1EC48F40 |
|  | 0000 | 1EC58C10 | 8001BFE4 | VIR0000 +4224 | 8040 | STILL HERE 1205122225 | 04000270 | 1EC48F60 |
|  | 0000 | 1EC58E88 | 8001BF68 | VIR0000 +41A8 | 8040 | STILL HERE 1205122225 | 04000170 | 1EC48F80 |
|  | 0000 | 0010E500 | 8001EE2A | VIR0000 +706A | 8040 | STILL HERE 1205122223 | 01000250 | 1EC48FA0 |
|  | 0000 | 0010E758 | 8001EE2A | VIR0000 +706A | 8040 | STILL HERE 1205122223 | 01000250 | 1EC48FC0 |
|  | 0000 | 0010E9B0 | 8001EE2A | VIR0000 +706A | 8040 | STILL HERE 1205122222 | 01000250 | 1EC48FE0 |
|  | 0000 | 0010EC08 | 8001EE2A | VIR0000 +706A | 8040 | STILL HERE 1205122222 | 01000250 | 1EC47000 |
|  | 0000 | 0010EE60 | 8001EE2A | VIR0000 +706A | 8040 | STILL HERE 1205122221 | 01000250 | 1EC47020 |
|  | 0000 | 0010F0B8 | 8001EE2A | VIR0000 +706A | 8040 | STILL HERE 1205122220 | 01000250 | 1EC47040 |
|  | 0000 | 0010F310 | 8001EE2A | VIR0000 +706A | 8040 | STILL HERE 1205122220 | 01000250 | 1EC47060 |
|  | 0000 | 0010F568 | 8001EE2A | VIR0000 +706A | 8040 | STILL HERE 1205122220 | 01000250 | 1EC47080 |

0000 0010F7C0 8001EE2A VIR0000 +706A 8040 STILL HERE 1205122219 01000250 1EC470A0

1

2

3

4

5

6

7

8

9

10

*Example of a memory allocataion history*

1. Line or terminal name for which memory allocation is performed. This information is omitted when the allocation relates VIRTEL itself.
2. Task number behind the allocation request.
3. Register 15 value.
4. Register 14 value.
5. Program name + offset of the origin request.
6. Memory allocation type. (8040 = GETMAIN).
7. Memory block state.
8. Time of the allocation.
9. Type and size of the allocation. The two first bytes represents the type of memory allocated (See [“Memory display](#_bookmark117) [of VIRTEL address space”, page 42](#_bookmark117) for a complete description of the memory block type.). The six last bytes represents the size of the memory block allocated.
10. Reserved for internal use.

**9.4. Memory Trace Overhead**

Tracing memory activity can produce an important overhead estimated to 20-30% of the activity. When using MEMTRACE command, the memory previously allocated to records history is released.

# 10. Trademarks

SysperTec, the SysperTec logo, syspertec.com and VIRTEL are trademarks or registered trademarks of SysperTec Communication Group, registered in France and other countries.

IBM, VTAM, CICS, IMS, RACF, DB2, MVS, WebSphere, MQSeries, System z are trademarks or registered trademarks of International Business Machines Corp., registered in United States and other countries.

Adobe, Acrobat, PostScript and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates. Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service names of others.

**10.1. Open Source Software**

The current VIRTEL Web Access product uses the following open source software: jQuery

Under MIT license

[https://jquery.org/license/](#_bookmark0).

StoreJson

Under MIT license [https://github.com/marcuswestin/store.js/commit/baf3d41b7092f0bacd441b768a77650199c25fa7](#_bookmark0).

jQuery\_UI

Under MIT license <http://en.wikipedia.org/wiki/JQuery_UI>.

# Index

|  |  |  |  |
| --- | --- | --- | --- |
| **Commands** [**, 13**](#_bookmark25) | **Line** |  | **Statistics** |
| KILL [, 17](#_bookmark44) |  | Monitoring [, 6](#_bookmark2) | File [, 30](#_bookmark97), [34](#_bookmark102) |
| LINE [, 15](#_bookmark33) |  | Start [, 15](#_bookmark37) | File status [, 34](#_bookmark103) |
| MEMDISPLAY , 0 |  | Status [, 7](#_bookmark4) | File switch [, 34](#_bookmark104) |
| MEMTRACE [, 45](#_bookmark126) |  | Stop [, 15](#_bookmark37) | Printing [, 34](#_bookmark106), [35](#_bookmark108), [40](#_bookmark112) |
| MSG [, 21](#_bookmark73) |  | Trace [, 18](#_bookmark49) | SMF [, 40](#_bookmark110) |
| NEW [, 20](#_bookmark69) |  | Virtual circuits status [, 10](#_bookmark15) | **Stop** |

RELAYS [, 15](#_bookmark35)

SILENCE [, 21](#_bookmark75)

SNAP [, 28](#_bookmark92)

SNAP80 [, 20](#_bookmark65)

SNAPMSG [, 19](#_bookmark63)

SNAPW [, 20](#_bookmark67)

START [, 15](#_bookmark37)

STAT [, 34](#_bookmark102)

STOP [, 15](#_bookmark37), [17](#_bookmark42)

TRACE [, 18](#_bookmark47)

TRACE / NOTRACE [, 18](#_bookmark51) VIRSV [, 20](#_bookmark71)

ZAP [, 21](#_bookmark77)

**Display**

Lines [, 15](#_bookmark31) Line Status [, 7](#_bookmark4) Relay [, 15](#_bookmark35)

**Memory**

Display content [, 16](#_bookmark39) Management [, 42](#_bookmark115)

Trace [, 45](#_bookmark126)

**Message suppression** [**, 21**](#_bookmark75) **Relay**

Display [, 15](#_bookmark35)

Snap [, 19](#_bookmark61)

Trace [, 18](#_bookmark47)

**Scenario**

Stop [, 17](#_bookmark44)

**SMF** [**, 40**](#_bookmark110)

**Snap**

Adjusting format [, 20](#_bookmark67) Diagnostics [, 19](#_bookmark57)

Message-triggered [, 19](#_bookmark63)

Relay [, 19](#_bookmark61)

System [, 19](#_bookmark59)

Terminal [, 19](#_bookmark61)

**Start**

Line [, 15](#_bookmark37)

Line [, 15](#_bookmark37)

Scenario [, 17](#_bookmark44)

VIRTEL [, 17](#_bookmark42)

**Terminal**

Snap [, 19](#_bookmark61)

Trace [, 18](#_bookmark47)

**Trace**

Activation [, 25](#_bookmark88) Deactivate all traces [, 19](#_bookmark55) Line [, 18](#_bookmark49)

List of active traces [, 19](#_bookmark53) Memory [, 45](#_bookmark126)

Relay [, 18](#_bookmark47)

Terminal [, 18](#_bookmark47)

**VIRSV**

Refreshing service program [, 20](#_bookmark71)

**VIRTEL**

Patch a program [, 21](#_bookmark77) Refreshing a program [, 20](#_bookmark69) Stopping [, 17](#_bookmark42)