



# VIRTEL Installation

---

## User's Guide

Version : 4.53

Release Date : 24 Sep 2014

Publication Date : 05/01/2016

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

<b>1.</b>	<b>Summary of Amendments.....</b>	<b>8</b>
1.1.	Virtel version 4.53 (24 Sep 2014) .....	8
1.2.	Virtel version 4.52 (4 Dec 2013) .....	9
1.3.	Virtel version 4.51 (18 Jul 2013) .....	9
1.4.	Virtel version 4.50 (30 Jun 2013) .....	10
1.5.	Virtel version 4.49 (30 Apr 2013) .....	10
1.6.	Virtel version 4.48 (27 Nov 2012) .....	11
1.7.	Virtel version 4.47 (21 May 2012) .....	12
<b>2.</b>	<b>Introduction .....</b>	<b>14</b>
2.1.	VIRTEL modules.....	14
2.2.	Required environment .....	15
2.2.1.	MVS environment.....	15
2.2.2.	VSE environment .....	15
2.2.3.	Workstation pre-requisites.....	15
<b>3.</b>	<b>Installing VIRTEL under MVS.....</b>	<b>16</b>
3.1.	Installation procedure .....	16
3.1.1.	Restoring from XMIT file .....	16
3.1.2.	Applying PTFs .....	19
3.1.3.	Upgrading a previous release .....	20
3.2.	Applying VWA User Interface Update .....	21
3.3.	Executing VIRTEL in an MVS environment.....	22
3.3.1.	Required and optional files for VIRTEL .....	22
3.3.2.	APF authorisation, userid, and priority .....	23
3.3.3.	Executing VIRTEL .....	23
3.3.4.	Stopping VIRTEL .....	23
3.4.	MVS installation check-list.....	23
<b>4.</b>	<b>Installing VIRTEL under VSE .....</b>	<b>26</b>
4.1.	Installation procedure .....	26
4.1.1.	Loading the installation jobs.....	26
4.1.2.	Defining the library .....	28
4.1.3.	Loading the executable modules.....	29
4.1.4.	Loading the source modules .....	29
4.1.5.	Defining the VIRARBO and VIRSWAP files .....	31

4.1.6.	Initialisation of the VIRARBO file .....	32
4.1.7.	Defining the VIRSTAT file .....	33
4.1.8.	Defining the VIRCMP3 file .....	33
4.1.9.	Defining the VIRCAPT file .....	34
4.1.10.	Defining the SAMPTRF file.....	34
4.1.11.	Defining the HTMLTRF file .....	35
4.1.12.	Loading the SAMPTRF file .....	36
4.1.13.	Defining the VIRHTML file .....	36
4.1.14.	Assembling the VIRTCT.....	37
4.1.15.	Assembling the MODVIRT mode table .....	37
4.1.16.	Updating the VIRARBO file (ARBOLOAD).....	37
4.1.17.	Cataloging the VTAM application book.....	38
4.1.18.	Defining the CICS resources.....	39
<b>4.2.</b>	<b>Applying VWA User Interface Update .....</b>	<b>40</b>
<b>4.3.</b>	<b>Executing VIRTEL in a VSE environment .....</b>	<b>40</b>
4.3.1.	Specifying the TCP/IP partition.....	41
4.3.2.	Stopping VIRTEL .....	41
<b>4.4.</b>	<b>Applying PTFs .....</b>	<b>41</b>
4.4.1.	Loading from a 3480 cartridge.....	41
4.4.2.	Application of the PTFs.....	41
<b>5.</b>	<b>VTAM parameters .....</b>	<b>42</b>
5.1.	VTAM parameters.....	42
5.2.	Definition of the VIRTEL primary ACB .....	42
5.3.	Defining the VTAM application relays .....	42
5.4.	MODETAB for Minitel, X25, and APPC.....	43
5.5.	USSTAB for Minitels and PC's.....	44
5.6.	CICS definitions .....	44
5.6.1.	VIRTEL Web Access Terminals .....	44
5.6.2.	Minitel Terminals.....	45
<b>6.</b>	<b>VIRTCT .....</b>	<b>48</b>
6.1.	Introduction .....	48
6.2.	Parameters of the VIRTCT .....	48
6.2.1.	ACCUEIL parameter.....	48
6.2.2.	ADDR1 parameter.....	49
6.2.3.	ADDR2 parameter.....	49
6.2.4.	AIC parameter.....	49
6.2.5.	ANNUL parameter.....	49
6.2.6.	APPLID parameter.....	49
6.2.7.	APPSTAT parameter.....	50
6.2.8.	ARBO parameter .....	50
6.2.9.	BATCH1 parameter.....	50
6.2.10.	BATCH2 parameter.....	51
6.2.11.	BFVSAM parameter.....	51
6.2.12.	BUFDATA parameter.....	51
6.2.13.	BUFSIZE parameter .....	51
6.2.14.	CHARSET parameter.....	51
6.2.15.	CODE parameter .....	52

6.2.16.	COMPANY parameter.....	52
6.2.17.	COMPR3 parameter.....	52
6.2.18.	CORRECT parameter.....	52
6.2.19.	COUNTRY parameter.....	52
6.2.20.	CRYPT1 parameter.....	53
6.2.21.	CRYPT2 parameter.....	55
6.2.22.	DEFENTR parameter.....	55
6.2.23.	DEFUTF8 parameter.....	55
6.2.24.	DIRECT parameter.....	56
6.2.25.	DONTSWA parameter (MVS only).....	57
6.2.26.	EXIT1 parameter.....	57
6.2.27.	EXIT2 parameter.....	57
6.2.28.	EXIT3 parameter.....	57
6.2.29.	EXIT4 parameter.....	57
6.2.30.	EXIT5 parameter.....	57
6.2.31.	EXIT6 parameter.....	58
6.2.32.	EXIT7 parameter.....	58
6.2.33.	EXIT8 parameter.....	58
6.2.34.	EXPIRE parameter.....	58
6.2.35.	FASTC parameter.....	58
6.2.36.	FCAPT parameter.....	58
6.2.37.	FCMP3 parameter.....	59
6.2.38.	GATE parameter.....	59
6.2.39.	GMT parameter.....	59
6.2.40.	GRNAME parameter.....	60
6.2.41.	GTLOAD parameter.....	60
6.2.42.	GTPRFE1 parameter.....	60
6.2.43.	GTPRFE2 parameter.....	60
6.2.44.	GTVSAM parameter.....	61
6.2.45.	GTVSKIP parameter.....	61
6.2.46.	GUIDE parameter.....	61
6.2.47.	HTFORWD parameter.....	61
6.2.48.	HTHEADR parameter.....	61
6.2.49.	HTMINI parameter.....	62
6.2.50.	HTPARM parameter.....	62
6.2.51.	HTSET1 to HTSET4 parameters.....	62
6.2.52.	HTVSAM parameter.....	63
6.2.53.	IBERTEX parameter.....	63
6.2.54.	IGNLU parameter.....	63
6.2.55.	LANG parameter.....	63
6.2.56.	LICENCE parameter.....	64
6.2.57.	LOCK parameter.....	64
6.2.58.	MARK parameter.....	64
6.2.59.	MAXAPPL parameter.....	64
6.2.60.	MEMORY parameter.....	64
6.2.61.	MINITEL parameter.....	65
6.2.62.	MQ1 parameter.....	65
6.2.63.	MQ2 parameter.....	65
6.2.64.	MULTI parameter.....	65
6.2.65.	NBCVC parameter.....	66
6.2.66.	NBDYNAM parameter.....	66
6.2.67.	NBTERM parameter.....	66
6.2.68.	NUMTASK parameter.....	66

6.2.69.	OTMAPRM parameter.....	66
6.2.70.	OSCORE parameter .....	67
6.2.71.	PACKET parameter .....	67
6.2.72.	PASSTCK parameter.....	67
6.2.73.	PREZ900 parameter .....	67
6.2.74.	PRFSECU parameter .....	67
6.2.75.	PWPROT parameter .....	68
6.2.76.	RACAPPL parameter.....	68
6.2.77.	RAPPL parameter.....	68
6.2.78.	REALM parameter.....	69
6.2.79.	REPET parameter .....	69
6.2.80.	RESO parameter.....	69
6.2.81.	RETOUR parameter .....	69
6.2.82.	RNODE parameter.....	70
6.2.83.	SECUR parameter.....	70
6.2.84.	SILENCE parameter .....	71
6.2.85.	SNAPW parameter .....	71
6.2.86.	SOMMR parameter .....	71
6.2.87.	STATDSN parameter .....	71
6.2.88.	STATS parameter .....	71
6.2.89.	STRNO parameter .....	72
6.2.90.	SUITE parameter .....	72
6.2.91.	SWAP parameter.....	73
6.2.92.	SYSPLUS parameter .....	73
6.2.93.	TCP1 parameter .....	73
6.2.94.	TCP2 parameter .....	74
6.2.95.	TIMEOUT parameter.....	74
6.2.96.	TIMERQS parameter.....	74
6.2.97.	TITRE1 parameter .....	74
6.2.98.	TITRE2 parameter .....	75
6.2.99.	TRACALL parameter .....	75
6.2.100.	TRACBIG parameter .....	75
6.2.101.	TRACEB parameter.....	75
6.2.102.	TRACEOJ parameter .....	75
6.2.103.	TRACEON parameter.....	76
6.2.104.	TRACTIM parameter.....	76
6.2.105.	TRAN parameter .....	76
6.2.106.	UFILE1 to UFILE20 parameters.....	76
6.2.107.	VIRSECU parameter.....	77
6.2.108.	VIRSV1 parameter.....	77
6.2.109.	VSAMTYP parameter.....	77
6.2.110.	VTKEYS parameter .....	78
6.2.111.	VTOVER parameter .....	78
6.2.112.	XM1 parameter.....	78
6.2.113.	XM2 parameter.....	78
6.2.114.	ZAPH parameter.....	79
<b>6.3.</b>	<b>Additional parameters for VSAM files.....</b>	<b>79</b>
<b>6.4.</b>	<b>Additional parameters for batch files.....</b>	<b>79</b>
<b>6.5.</b>	<b>How to share VSAM files between multiple instances of VIRTEL .....</b>	<b>80</b>
<b>6.6.</b>	<b>Example of the VIRTCT .....</b>	<b>80</b>
<b>6.7.</b>	<b>Assembling the VIRTCT.....</b>	<b>82</b>
6.7.1.	MVS example .....	82

6.7.2.	VSE example .....	82
6.8.	Dynamic VIRTCT overrides.....	83
6.9.	Applying patches via the VIRTCT.....	84
<b>7.</b>	<b>VIRCONF .....</b>	<b>85</b>
7.1.	Introduction .....	85
7.2.	JCL.....	85
7.2.1.	Creating and uploading a new VIRARBO file.....	85
7.2.2.	Updating a VIRARBO file .....	86
7.2.3.	Unloading a VIRARBO file .....	87
7.2.4.	Verify control card syntax.....	87
7.2.5.	Multi-language support .....	88
7.3.	VIRCONF control cards.....	89
7.3.1.	VIRCONF control card syntax.....	89
7.3.2.	APPLIC .....	89
7.3.3.	DELETE .....	90
7.3.4.	DEPT.....	90
7.3.5.	ENTRY.....	90
7.3.6.	INDEX .....	91
7.3.7.	LINE .....	91
7.3.8.	NODE.....	92
7.3.9.	PC .....	92
7.3.10.	PROFILE .....	92
7.3.11.	RESOURCE .....	92
7.3.12.	RULE.....	93
7.3.13.	SERVER .....	94
7.3.14.	SSERV .....	94
7.3.15.	SUBDIR .....	95
7.3.16.	TERMINAL .....	96
7.3.17.	TRANSACT .....	96
7.3.18.	UPDATE .....	97
7.3.19.	USER.....	97
<b>8.</b>	<b>Security.....</b>	<b>98</b>
8.1.	How to activate RACF security .....	98
8.1.1.	Modify the VIRTCT.....	98
8.1.2.	Add RACF definitions .....	98
8.1.3.	Refine RACF definitions .....	99
8.2.	How to activate ACF2 security .....	99
8.2.1.	Modify the VIRTCT.....	99
8.2.2.	Determine the ACF2 resource type.....	100
8.2.3.	Add ACF2 definitions .....	100
8.3.	How to activate Top Secret (TSS) security .....	101
8.3.1.	Modify the VIRTCT.....	101
8.3.2.	Add TSS definitions.....	101
<b>9.</b>	<b>Trademarks .....</b>	<b>106</b>
9.1.	Open source software .....	106

# 1. Summary of Amendments

## 1.1. Virtel Version 4.53 (24 Sep 2014)

### VIRTEL Web Integration:

- Additional commands for scenarios:
  - ACTION\$ REFRESH-TERMINAL
  - HANDLE\$ ABEND
  - HANDLE\$ LOOP
  - PERFORM\$ subroutine
  - LABEL\$

### Miscellaneous:

- Increased CISIZE for the data portion of the TRSF's files
- Modification of SHR options for ARBO file
- SMF support for statistics records
- Support of multi lines for WTO
- TRACTIM=CPU CPU time in snap
- Allows to specify a country code in the VIRTCT VIRSV1= second sub-parameter
- Allows BLDVRP hiperspace option in VIRTCT
- Miscellaneous editorial corrections and enhancements
- Additional system commands:
  - SNAPMSG command to trap VIRHT6xE message
- Messages:
  - Message "VIR0000I acbname STARTED AT dd/mm/aa hh:mm:hh, VERSION x.xx" is now referenced under message number VIR0099I without any other modification.



## 1.2. Virtel Version 4.52 (4 Dec 2013)

### VIRTEL Universal Protocol:

- OTMAPRM parameter to specify security parameters for RESUME TPIPE
- Modifiable exit name for RESUME TPIPE
- Selection of transaction name for RESUME TPIPE

### VIRTEL Web Access:

- Macro management enhancements
- Size limit for IND\$FILE upload
- Support for BLINK attribute in IE8+, Firefox, and Chrome
- Additional HTML tags:
  - VALUE-OF URL and QUERY

### VIRTEL Web Modernisation:

- Timeout specifiable for SEND\$ TO-LINE
- Additional commands for scenarios:
  - COPY\$ SYSTEM-TO-VARIABLE URL and QUERY
  - SEND\$ MAXTIME

### Miscellaneous:

- VIRCONF documentation in English
- Modernised configuration interface
- Support for pre-zSeries processors
- Unique identification for VIRTEL startup message
- Customizable VIRTEL application name for RACF (RACAPPL)
- Miscellaneous editorial corrections and enhancements

## 1.3. Virtel Version 4.51 (18 Jul 2013)

### VIRTEL Universal Protocol:

- Protocol ICONNECT (RESUME TPIPE) for communication with IMS

### VIRTEL Web Access:

- Additional HTML tags:
  - NAME-OF VIRTEL-VERSION

### VIRTEL Web Modernisation:

- Additional commands for scenarios:
  - COPY\$ SYSTEM-TO-VARIABLE VIRTEL-VERSION

### Miscellaneous:

- Open and close printers on demand
- Repeating terminal definitions in hexadecimal and alphanumeric
- Display printer names by F VIRTEL,L=line,D command

- New command F VIRTEL,RELAYS
- Miscellaneous editorial corrections and enhancements

### 1.4. Virtel Version 4.50 (30 Jun 2013)

#### VIRTEL Web Access:

- Passticket support
- New Web Access Settings:
  - Shift+Enter, Ctrl+Enter
  - Line spacing, Character spacing
- Ctrl-A to mark whole 3270 screen for copy
- Support for 3270 Graphic Escape characters
- Downloadable fonts
- Print SCS-to-PDF
- Query support for SCS printers
- Printer autoconnect
- Miscellaneous ergonomic enhancements

#### VIRTEL Web Modernisation:

- Additional commands for scenarios:
  - CONVERT\$ EBCDIC-TO-UPPERCASE, EBCDIC-TO-LOWERCASE
  - COPY\$ PASSTICKET

#### MQSeries:

- Unique correlator for MQ requests
- Message type REPLY for MQ responses

#### Miscellaneous:

- Miscellaneous editorial corrections and enhancements

### 1.5. Virtel Version 4.49 (30 Apr 2013)

#### VIRTEL Web Access:

- Codepages 0037 and 1047 included as standard
- New Web Access Settings:
  - Adapt font size ratio
  - Additional keyboard remappings for Alt+Ins, Alt+Home, and Alt+F1
  - ChgCur key to change cursor shape
- Settings page in French and German
- Administrator can hide specific settings
- Additional HTML tags:
  - CREATE-VARIABLE-IF TRACING-SCENARIO
  - SET-HEADER

- Allow Virtel session code to be stored in cookie
- Custom hotspot recognition
- Custom logo for Web Access and Application menus
- Drag and drop upload summary report
- Miscellaneous ergonomic enhancements

VIRTEL Web Modernisation:

- Additional commands for scenarios:
  - COPY\$ SYSTEM-TO-VARIABLE USER-SECURITY-PROFILE
  - DEBUG\$
  - SET\$ SIGNON
  - VIRSV\$ TRANSACTION OPTION=CLOSE

SYSPLEX support:

- Allow read-only sharing of VSAM files

Miscellaneous:

- Miscellaneous editorial corrections and enhancements

## 1.6. Virtel Version 4.48 (27 Nov 2012)

VIRTEL Universal Protocol:

- Menu program VIR0021J

VIRTEL Web Access:

- Connection of non-predefined VTAM LU names
- Support for expired passwords
- Site-specific Javascript (custom.js)
- REALM parameter to reduce multiple signon
- Codepage 1047 for C programming
- Codepage override by URL
- New Web Access Settings:
  - End key
  - Remap caret to logical not
- Additional HTML tags:
  - CREATE-VARIABLE-FROM  
allows a rectangle to be copied from the 3270 screen
- Miscellaneous ergonomic enhancements

VIRTEL Web Modernisation:

- Support for scenarios stored in VSAM
- Additional commands for scenarios:
  - ACTION\$ PROCESS-RESPONSE
  - COPY\$ VARIABLE-TO-SYSTEM PASSWORD

- IF\$ SCREEN-IS-BLANK
- IF\$ SCREEN-IS-UNFORMATTED

### VIRTEL Web Integration:

- New programs CALL VIRSETAI, VIRSETVI for IMS

### SYSPLEX support:

- SYSCONE parameter in LU names
- Dynamic VIRTCT overrides

### Miscellaneous:

- Restart VIRSV service by console command
- Patch application via the VIRTCT
- Miscellaneous editorial corrections and enhancements

## 1.7. Virtel Version 4.47 (21 May 2012)

### VIRTEL Universal Protocol:

- Native TCP/IP protocol

### VIRTEL Web Access

- Support for 3270 FieldMark and Dup characters
- Keystroke buffering
- New Web Access Settings:
  - Move cursor on activate
  - Omit nulls from input
  - Select word by double-click
  - Highlight input fields
  - Keep keypad and macro pad open
  - Field mark and Dup
- Additional HTML tags:
  - DELETE-ALL-VARIABLES  
allows the variable pool to be reset from within page template
  - SET-INITIAL-TIMEOUT  
allows an initial timeout to be specified within a page
  - SET-LOCAL-OPTIONS JSON-ESCAPES  
allows generation of JSON page templates
  - SET-LOCAL-OPTIONS TRACE-LINE, TRACE-RELAY  
allows traces to be activated from a page template
- Upload by drag and drop (Chrome only)
- Save and restore file transfer parameters
- Long polling reduces load on IP network
- Customizable user help page for Web Access

- Site customization of colors and logo (custom.css)
- Miscellaneous ergonomic enhancements

VIRTEL Web Modernisation:

- Screen redesigner upgrade
- Additional commands for scenarios:
  - CASE\$ VARIABLE
  - COPY\$ LIST-TO-VARIABLE
  - COPY\$ VARIABLE-TO-SYSTEM NAME-OF DIRECTORY
  - FILTER\$ VARIABLE-TO-VARIABLE
  - MAP\$ EXECUTE and RETURN\$

Miscellaneous:

- Logon screen for VTAM applications
- Latin-2 translate tables for Eastern European countries
- Miscellaneous editorial corrections and enhancements

## 2. Introduction

### 2.1. VIRTEL Modules

The functionality of VIRTEL is divided into components known as “modules”. The following is a list of the VIRTEL modules:

- The VIRTEL base kernel
- VIRTEL Multi-Session
- VIRTEL Security
- VIRTEL Web Access
- VIRTEL Web Modernisation
- VIRTEL Web Integration
- VIRTEL Application - Application
- VIRTEL XOT
- VIRTEL Minitel / 3270
- VIRTEL Compression
- VIRTEL PC / VT100
- VIRTEL Network Manager (VTAM console)
- VIRTEL Incoming calls SMTP
- VIRTEL Incoming calls Minitel
- VIRTEL Incoming calls PC
- VIRTEL Incoming calls VTxxx
- VIRTEL Outgoing calls SMTP
- VIRTEL Outgoing calls Videotex 3270
- VIRTEL Outgoing calls 3174 switched X25
- VIRTEL Outgoing calls VTxxx
- VIRTEL VIRAPI, APPC
- VIRTEL LECAM
- VIRTEL VIRPASS

The VIRTEL product contains support for the base kernel and all modules. The functionality of each module is activated either by setting specific parameters in the VIRTCT or by the activation of appropriate configuration definitions in the VIRARBO file.

Please refer to your license agreement for the particular terms and conditions under which you are authorised to use the various VIRTEL modules.

## 2.2. Required Environment

VIRTEL operates in the MVS or VSE environments. Throughout the VIRTEL documentation, the term “MVS” should be understood to include OS/390 and z/OS, and the term “VSE” should be understood to include VSE/ESA and z/VSE.

### 2.2.1. MVS environment

In the MVS environment, VIRTEL runs under the OS/390 or z/OS operating systems. If the VIRTEL MQ interface is used, then MQSeries Version 6 or later is required. Support for the cryptographic functions of VIRTEL requires ICSF Version HCR7740 or later.

### 2.2.2. VSE environment

In the VSE environment, VIRTEL runs under the VSE/ESA or z/VSE operating systems. TCP/IP access (XOT, VIRTEL Web Access) requires VSE/ESA 2.5.1 or later, or any version of z/VSE.

### 2.2.3. Workstation pre-requisites

VIRTEL Web Access requires a standard web browser on the user’s workstation. Supported browsers include:

- Microsoft Internet Explorer Version 6 or above
- Firefox Version 15 or above (for Windows 7 or Vista)
- Firefox Version 17 or above (for Windows XP)
- Chrome Version 23 or above
- Opera Version 15 or above
- Safari Version 5 or above

VIRTEL Web Access requires JavaScript to be enabled in the browser.

## 3. Installing VIRTEL under MVS

### 3.1. Installation Procedure

In the MVS environment, VIRTEL is delivered as a zipped XMIT file containing the VIRTEL datasets compressed in DF/DSS dump format. The following sections provide details of the installation method.

A quick “installation check-list” to start the VIRTEL Web Access function for MVS can be found at the end of this section.

#### 3.1.1. Restoring from XMIT file

All the VSAM and non-VSAM datasets required for the installation of VIRTEL are contained in a zipped XMIT file which can be downloaded from the Syspertec file server. The size of the zipped file is approximately 2MB. Two JCL files (\$ALOCDSU and \$RESTDSU) are also included in the zip file. The procedure for obtaining and uploading the file is as follows:

##### 3.1.1.1. STEP 1

Login to the Syspertec file server <http://ftp.syspertec.com> using the userid and password supplied to you by Syspertec. Navigate to the “Public” – “VIRTEL 4.53” – “Products” folder and download the virtel453mvs.zip file. Unzip this file into a folder on your workstation.

##### 3.1.1.2. STEP 2

Run the job \$ALOCDSU to allocate a sequential file named userid.TRANSFER.XMIT with DCB attributes (RECFM=FB, LRECL=80):

```
//SPAL0DSU JOB 1,MSGCLASS=X,CLASS=A,NOTIFY=&SYSUID
//*-----*
//*--* BINARY FILE TRANSFER - STEP NO.1 *--*
//*--* *--*
//*--* Function : Allocate a sequential XMIT type file *--*
//*--* *--*
```



```

/*--* Following step $RESTDSU *--*
/*--* *--*
/*-----*
//          SET TYPE=CYL                /* TYPE ALLOC */
//          SET ALLOCPRI=4              /* PRIM ALLOC */
//          SET ALLOCSEC=1              /* SECO ALLOC */
//          SET VOLM=SPT001             /* VOLUME */
//          SET UNIT=3390                /* DISK UNIT */
/*-----*
/* DELETE OLD .XMIT file *
/*-----*
//STEP1 EXEC PGM=IKJEFT01,PARM='DEL ''&SYSUID..TRANSFER.XMIT'''
//SYSTSPRT DD SYSOUT=*
//SYSOUT DD *
//SYSTSIN DD *
PROF /* POUR GENERER CC=0 */
/*
/*-----*
/* Allocate new reception .XMIT file *
/*-----*
//STEP2 EXEC PGM=IEFBR14
//SYSOUT DD *
//SYSUT2 DD DSN&SYSUID..TRANSFER.XMIT,
//          UNIT=&UNIT,VOL=SER=&VOLM,DISP=(NEW,CATLG),
//          SPACE=(&TYPE,(&ALLOCPRI,&ALLOCSEC)),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120,DSORG=PS)
/*
/*-----*
/*--* BINARY FILE TRANSFER - STEP NO.2 *--*
/*--* Make a binary transfer of the given file *--*
/*--* BIN *--*
/*--* PUT filename.xmit TRANSFER.XMIT *--*
/*-----*

```

*JCL for allocating an XMIT file (MVS)*

The parameters SET VOLM=SPT001 and SET UNIT=3390 at the start of this job should be changed as appropriate to match the volume on which the userid.TRANSFER.XMIT dataset is to be allocated.

### 3.1.1.3. STEP 3

Using FTP or IND\$FILE, upload the file virtel453mvs.xmit to the host transfer file created in step 1. It is very important to ensure that the upload is performed in binary mode. The following is an example of an FTP session to perform the upload:

```

C:\temp>ftp 192.168.0.1
Connected to 192.168.0.1.
220-FTPD1 IBM FTP CS V1R4 at ZOS1.COMPANY.COM, 08:41:36 on 2004-05-24.
220 Connection will close if idle for more than 5 minutes.
User (192.168.0.1:(none)): sptuser
331 Send password please.
Password:
230 SPTUSER is logged on. Working directory is "SPTUSER.".
ftp> bin
200 Representation type is Image
ftp> put virtel453mvs.xmit TRANSFER.XMIT
200 Port request OK.
125 Storing data set SPTUSER.TRANSFER.XMIT
250 Transfer completed successfully.
ftp: 4067120 bytes sent in 5,59Seconds 727,83Kbytes/sec.
ftp> quit
221 Quit command received. Goodbye.
C:\temp>

```

*Figure 2-2 FTP session for uploading an XMIT file (MVS)*

**3.1.1.4. STEP 4**

Run the job \$RESTDSU to unpack the transfer file and to restore the VIRTEL datasets by means of the ADRDSSU utility program:

```
//SPRESDSU JOB 1,MSGCLASS=X,CLASS=A,NOTIFY=&SYSUID
//*-----*
//*--* Binary File Transfer - STEP No 3 *--*
//*--* *--*
//*--* Function : Reception and reload of the files *--*
//*--* *--*
//*--* Replace '?????' by target volume serial number *--*
//*--* Replace 'yourqual' by target DSN high-level qualifier *--*
//*-----*
//*-----*
//* Reception of the .XMIT File *
//*-----*
//BATCHTS EXEC PGM=IKJEFT1A,REGION=4M
//SYSPRINT DD SYSOUT=*
//SYSTSPRT DD SYSOUT=*
//XMITFILE DD DSN=&SYSUID..TRANSFER.XMIT,DISP=OLD
//SYSTSIN DD *
RECEIVE INFILE(XMITFILE)
DA(TRANSFER.DSSDUMP)
//*
//*-----*
//* Reload of the initial files *
//*-----*
//DSSREST EXEC PGM=ADRDSSU,REGION=6M,COND=(0,NE)
//SYSPRINT DD SYSOUT=*
//DUMPFIL DD DSN=&SYSUID..TRANSFER.DSSDUMP,DISP=(OLD,DELETE)
RESTORE -
DS(INCLUDE(SPRODUI.VIRTEL.BASE*.*)) -
OUTDYNAM(?????,3390) /* <==== VOLUME, UNIT ===== */ -
RENAMEUNC( -
(SPRODUI.VIRTEL.BASE453.LOADLIB, -
yourqual.VIRT453.LOADLIB), -
(SPRODUI.VIRTEL.BASE453.MACLIB, -
yourqual.VIRT453.MACLIB), -
(SPRODUI.VIRTEL.BASE453.SAMPLIB, -
yourqual.VIRT453.SAMPLIB), -
(SPRODUI.VIRTEL.BASE453.SERVLIB, -
yourqual.VIRT453.SERVLIB), -
(SPRODUI.VIRTEL.BASE453.DBRMLIB, -
yourqual.VIRT453.DBRMLIB), -
(SPRODUI.VIRTEL.BASE453.CNTL, -
yourqual.VIRT453.CNTL), -
(SPRODUI.VIRTEL.BASE453.SAMP.TRSF, -
yourqual.VIRT453.SAMP.TRSF), -
(SPRODUI.VIRTEL.BASE453.CONFGEN.MACLIB, -
yourqual.VIRT453.CONFGEN.MACLIB), -
(SPRODUI.VIRTEL.BASE453.FA29API.MACLIB, -
yourqual.VIRT453.FA29API.MACLIB), -
(SPRODUI.VIRTEL.BASE453.SCRNAPI.MACLIB, -
yourqual.VIRT453.SCRNAPI.MACLIB), -
(SPRODUI.VIRTEL.BASE453.VIRAPI.MACLIB, -
yourqual.VIRT453.VIRAPI.MACLIB), -
(SPRODUI.VIRTEL.BASE453.ARBO, -
yourqual.VIRT453.ARBO), -
(SPRODUI.VIRTEL.BASE453.CAPT, -
yourqual.VIRT453.CAPT), -
(SPRODUI.VIRTEL.BASE453.CMP3, -
yourqual.VIRT453.CMP3), -
(SPRODUI.VIRTEL.BASE453.HTML, -
yourqual.VIRT453.HTML), -
(SPRODUI.VIRTEL.BASE453.HTML.TRSF, -
yourqual.VIRT453.HTML.TRSF), -
(SPRODUI.VIRTEL.BASE453.PLUG.TRSF, -
```

```

        yourqual.VIRT453.PLUG.TRSF), -
        (SPRODUIT.VIRTEL.BASE453.SWAP, -
        yourqual.VIRT453.SWAP), -
        (SPRODUIT.VIRTEL.BASE453.STAT, -
        yourqual.VIRT453.STAT), -
        ) -
/* NULLSTORCLAS BYPASSACS(**) */ /* <==== SMS OVERRIDE ===== */ -
/* ADMIN TOL(ENQF) */ /* <==== OPTIONAL ===== */ -
/* REPLACE SHR */ /* <==== SI EXISTE DEJA === */ -
        CATALOG INDD(DUMPFIL)
//*
//

```

*JCL for restoring from an XMIT file (MVS)*

The following changes should be made to this job before submitting it:

- If the VIRTEL datasets are not to be managed by SMS, alter the statement OUTDYNAM(?????,3390) to specify the volume on which the datasets are to be allocated.
- If the VIRTEL datasets are to be managed by SMS, remove the NULLSTORCLAS BYPASSACS(\*\*) statement and replace it by STORCLAS(classname) where classname is the name of the SMS storage class on which the VIRTEL datasets are to be allocated. Do not delete the OUTDYNAM parameter, ADRDSSU requires it even though its value is ignored for SMS.
- In the RENAMEUNC parameter, replace yourqual by the high-level qualifiers to be used for your VIRTEL datasets.
- The ADMIN and TOL(ENQF) parameters may be uncommented if you are authorized to the necessary STGADMIN profiles.

### 3.1.2. Applying PTFs

As a general rule the application of PTFs is necessary and recommended. PTFs are maintenance files which must be applied to the VIRTEL LOADLIB to correct problems which have been discovered subsequent to the building of the VIRTEL 4.53 release, or to add new function which will be included as standard in the next release. A second type of PTF consists of elements such as HTML pages, style sheets, and JavaScript files, which must be uploaded into the VIRTEL directories in the SAMPTRSF VSAM file. This type of PTF may sometimes be supplied as a complete replacement for the SAMPTRSF file in the form of a DF/DSS dump in XMIT format.

#### 3.1.2.1. Obtaining the PTFs

To download PTFs from the Syspertec file server, use your web browser to login to the file server as described 13, navigate to the “Public” – “VIRTEL 4.53” – “PTFS for version 4.53” folder, and download the allptfs-mvs453.txt file. If the file does not exist, then there are no PTFs to be applied.

Alternatively, you may receive the allptfs-mvs453.txt file by e-mail from Syspertec support.

#### 3.1.2.2. Uploading the PTFs

The allptfs-mvs453.txt file should be uploaded in text format to member PTF453MV of the VIRTEL CNTL library.

For PTFs which contain elements to be uploaded to VIRTEL, first unzip the elements to a directory on your workstation. Then use the “Upload” link from the VIRTEL Web Access page at <http://n.n.n.n:41001> to upload the elements one by one to the W2H-DIR directory.

In the case of a PTF containing a replacement SAMPTRSF file in DF/DSS XMIT format, use the procedure previously described (\$ALOCDSU and \$RESTDSU) to upload the file in binary and retrieve the SAMPTRSF VSAM file.

### 3.1.2.3. Applying the PTFs

The recovered PTFs are applied to the VIRTEL LOADLIB by using AMASPZAP with the IGNIDRFULL parameter. The ZAPJCL member in the VIRTEL CNTL library (shown below) performs the apply. This job should complete with return code 0000 or 0004.

```
//VIRPTF   JOB 1,ZAPJCL,CLASS=A,MSGCLASS=X,NOTIFY=&SYSUID
//*
//*       PTF à APPLIQUER
//*
// SET LOAD=yourqual.VIRT453.LOADLIB
// SET CNTL=yourqual.VIRT453.CNTL
// SET MEMBER=PTF453MV
//*
//PTFZAP   EXEC PGM=AMASPZAP,PARM='IGNIDRFULL'
//SYSPRINT DD SYSOUT=*
//SYSLIB   DD DSN=&LOAD,DISP=SHR
//SYSIN    DD DSN=&CNTL(&MEMBER),DISP=SHR
```

*Member ZAPJCL for applying PTFs (MVS)*

### 3.1.2.4. Restarting VIRTEL and validation of PTF level

VIRTEL must be stopped and restarted to allow the newly-applied PTFs to take effect. The list of PTFs applied is displayed near the beginning of the SYSMSG LG dataset during VIRTEL startup by message VIR0018I, as shown in the following example:

```
VIR0018I 3876,3876A,3882,3902,3904,3906,3928,3934
```

*Validation of the VIRTEL PTF level*

## 3.1.3. Upgrading a previous release

---

### 3.1.3.1. Datasets to be upgraded

If you already have a previous release of VIRTEL installed (version 4.00 or later) then you only need the datasets shown in the figure below:

```
yourqual.VIRT453.LOADLIB
yourqual.VIRT453.MACLIB
yourqual.VIRT453.SAMPLIB
yourqual.VIRT453.SERVLIB
yourqual.VIRT453.DBRMLIB
yourqual.VIRT453.CNTL
yourqual.VIRT453.SAMP.TRSF
yourqual.VIRT453.CONFGEN.MACLIB
yourqual.VIRT453.FA29API.MACLIB
yourqual.VIRT453.SCRNAPI.MACLIB
yourqual.VIRT453.VIRAPI.MACLIB
```

*Datasets upgraded during release change*

For the remaining datasets, shown in the figure below, you should continue to use your existing datasets, as these may contain customer-specific configuration information which you do not want to overwrite:

```
yourqual.VIRTnnn.ARBO
yourqual.VIRTnnn.CAPT
yourqual.VIRTnnn.CMP3
```

```
yourqual.VIRTnnn.HTML
yourqual.VIRTnnn.HTML.TRSF
yourqual.VIRTnnn.SWAP
yourqual.VIRTnnn.STAT
```

*Datasets to be retained from previous release*

Note: it is also possible to copy your existing files into the files of the new release using IDCAMS REPRO (or by ARBOLOAD for the VIRARBO file).

### 3.1.3.2. Upgrade procedure

The procedure for upgrading from a previous release of VIRTEL (version 4.00 or later) is as follows. Customers upgrading from earlier releases of VIRTEL should contact Syspertec for technical support.

1. Upload and unpack the virtel453mvs.xmit file as described in the previous section.
2. Apply PTFs as described in the previous section.
3. Copy your VIRTCTnn from the old VIRTnnn.CNTL library to the new VIRT453.CNTL
4. Reassemble your VIRTCTnn module using the ASMTCT job in VIRT453.CNTL
5. If you have any scenario or user exit modules, copy them to the VIRT453.CNTL library and reassemble them using the ASMSCEN and ASMEXIT jobs respectively.
6. Add the new VIRT453.LOADLIB library to the system APF list in the MVS PARMLIB and use the SETPROG command to authorize the VIRT453.LOADLIB library.
7. Edit your VIRTEL procedure in the MVS PROCLIB, to ensure that the STEPLIB, DFHRPL, and SERVLIB DD statements reference the new VIRT453.LOADLIB, and that the SAMPTRSF DD statement references the new VIRT453.SAMP.TRSF dataset.
8. If upgrading from a version prior to VIRTEL 4.43, add a VIRTRACE DD statement to the VIRTEL procedure, as shown in the next section.
9. If you have modified the default values for the VIRTEL Web Access Settings (as described in the VIRTEL Web Access Guide), upload your customized w2hparm.js file into the CLI-DIR directory and check that the CLI-03P transaction (under the CLIWHOST entry point) references CLI-DIR in the "Application" field. If you do not find a CLI-03P transaction, run job CUSTCSS from the VIRTEL SAMPLIB.
10. Stop and restart VIRTEL.

## 3.2. Applying VWA User Interface Update

Under certain circumstances it may be necessary to apply maintenance in the form of User Interface Updates. These may be distributed either by e-mail, or available on Syspertec FTP Server.

An update is available as a ZIP file containing the cumulative days update the version. The file is represented in the form VirtelxxxUpdtnnnn.zip where xxx is the version of Virtel to which it relates and nnnn the reference of the update itself. Once unzipped, the file content is in the form of a tree where each folder contains one or more files grouped by category, the root contains a file named updtnnnn.txt which summarized the history of changes and any special instructions to operate.

Generally, the file still contains a sub directory named " W2H " whose content must be reloaded into the W2H-DIR using one of the methods described in section "Uploading HTML Pages" from document "Virtel Web Access User Guide".

**If you are installing a Virtel version 453, you must:**

- First, intall the update number virtel453updt5228.zip
- Then intall that last level of update available, even if the version number is greathe than 453.

### 3.3. Executing VIRTEL In An MVS Environment

VIRTEL can run as a JOB or as an STC. An example JCL procedure is contained in member VIRTEL4 of the VIRTEL SAMPLIB. If VIRTEL is to be run as an STC, this member must be copied into a system PROCLIB and renamed as VIRTEL:

```
//VIRTEL PROC QUAL=yourqual.VIRT453,
//*      QUALMQ=CSQ600,          <-- MQSeries qualifier
//      APPLID=,                <-- Default is in VIRTCT
//      TCT=01                  <-- Suffix of VIRTCT
//*-----*
//* PROCEDURE LANCEMENT VIRTEL *
//*-----*
//VIRTEL EXEC PGM=VIR6000,
//      TIME=1440,REGION=32M,
//      PARM=(&TCT,&APPLID)
//STEPLIB DD DSN=&QUAL..LOADLIB,DISP=SHR
//*      DD DSN=&QUALMQ..SCSQANLE,DISP=SHR
//*      DD DSN=&QUALMQ..SCSQAUTH,DISP=SHR
//DFHRPL DD DSN=&QUAL..LOADLIB,DISP=SHR
//*      DD DSN=GTM.MIRES.LOADLIB,DISP=SHR
//SERVLIB DD DSN=&QUAL..SERVLIB,DISP=SHR
//VIRARBO DD DSN=&QUAL..ARBO,DISP=SHR
//VIRSWAP DD DSN=&QUAL..SWAP,DISP=SHR
//VIRSTAT DD DSN=&QUAL..STAT,DISP=SHR
//*VIRCMP3 DD DSN=&QUAL..CMP3,DISP=SHR
//*VIRCAPT DD DSN=&QUAL..CAPT,DISP=SHR
//VIRHTML DD DSN=&QUAL..HTML,DISP=SHR
//SAMPTRSF DD DSN=&QUAL..SAMP.TRSF,DISP=SHR
//HTMLTRSF DD DSN=&QUAL..HTML.TRSF,DISP=SHR
//*PLUGTRSF DD DSN=&QUAL..PLUG.TRSF,DISP=SHR
//SYSOUT DD SYSOUT=*
//VIRLOG DD SYSOUT=*
//VIRTRACE DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
```

*VIRTEL started task JCL procedure (MVS)*

#### 3.3.1. Required and optional files for VIRTEL

- Files STEPLIB, DFHRPL are always required
- Files VIRARBO, VIRSWAP are always required
- File SERVLIB must be present if the VIRSV1 parameter is coded in the VIRTCT
- File VIRSTAT must be present if the parameter STATS=YES is coded in the VIRTCT
- File VIRCMP3 must be present if the parameter COMPR3=AUTO or COMPR3=FIXED is coded in the VIRTCT
- File VIRCAPT must be present if the parameter FCAPT=VIRCAPT is coded in the VIRTCT
- File VIRHTML must be present if the parameter HTVSAM=VIRHTML is coded in the VIRTCT (parameter required for clients wishing to use e-mail, VIRTEL Web Access, or the Videotex Plug-In function)
- Files SAMPTRSF, HTMLTRSF must be present if referenced by the parameters UFILEn (and their corresponding ACBs) in the VIRTCT (required for clients wishing to use VIRTEL Web Access functions)
- File PLUGTRSF must be present if referenced by a parameter UFILEn (and its corresponding ACB) in the VIRTCT (required for clients wishing to use the Videotex Plug-In function)
- Files SYSOUT, VIRLOG, VIRTRACE, SYSPRINT, SYSUDUMP are always required
- The libraries SCSQANLE, SCSQAUTH must be concatenated to the STEPLIB unless these libraries are in the system link list or LPA list (only for clients wishing to use VIRTEL with MQSeries)
- The CSF.SCSFMODE0 library must be concatenated to the STEPLIB or must be present in the system link list (only if the CRYPTn=(...,ICSF,...) parameter is coded in the VIRTCT)

### 3.3.2. APF authorisation, userid, and priority

VIRTEL must run from an APF-authorized library if either of the following is true:

- External security (RACF, TOP SECRET, or ACF2) is selected by means of the SECUR parameter of the VIRTCT
- VIRTEL is made non-swappable by means of the DONTSWA parameter of the VIRTCT

Normally VIRTEL is started in APF-authorized mode via the VIR6000 module, and in this case all of the libraries specified in the STEPLIB and DFHRPL concatenations must be APF-authorized. For certain specialised applications (Videotex server), the DFHRPL concatenation may include screen image libraries which cannot be APF authorised. In this case it is possible to start VIRTEL via the module VIR0APF which can be isolated in an authorised library. In this way, the other libraries declared in DFHRPL do not necessarily need to be APF-authorized.

VIRTEL must be run under a userid which has an OMVS segment defined in its profile. If VIRTEL is started as an STC, define a profile in the RACF STARTED class (or equivalent if using another security product) to assign the VIRTEL STC to the appropriate userid.

It is necessary for VIRTEL to run at the same priority as VTAM and TCP/IP. This is usually done by assigning VIRTEL to service class SYSSTC in the workload manager. It is also recommended that VIRTEL run non swappable (DONTSWA=YES in the VIRTCT).

### 3.3.3. Executing VIRTEL

VIRTEL is started by executing the command S VIRTEL from the system console. Message VIR0000I indicates that the product started properly.

### 3.3.4. Stopping VIRTEL

VIRTEL may be stopped by issuing the following command: P VIRTEL

## 3.4. MVS Installation Check-list

Here is a standard “check-list” to start the WEB to HOST VIRTEL function:

1. Download the following files from our FTP server (<http://ftp.sypertec.com>)
  - Vrtel453mvs.zip.
  - allptfs-mvs453.txt if available.
  - vrtel453updtntnnn.zip if available.
2. Run job \$ALOCDSU to create the TRANSFER.XMIT file.
3. Upload the vrtel453mvs.xmit file to the TRANSFER.XMIT file in binary mode.
4. Edit job \$RESTDSU specifying the high-level qualifiers and SMS or volume serial information for the VIRTEL datasets. Run job \$RESTDSU to create the VIRTEL datasets yourqual.VIRT453.xxxxxx
5. Apply the PTFs in the allptfs-mvs453.txt file using job ZAPJCL in the VIRTEL CNTL library. If this file does not exist, skip this step.
6. Use the SETPROG APF command to add the VIRTEL LOADLIB to your system APF authorized program library list:  
SETPROG APF,ADD,DSN=yourqual.VIRT453.LOADLIB,VOL=volser

7. Edit member VIRTCT01 in the VIRTEL CNTL library :

1. Set the APPLID= parameter to the VTAM ACBNAME you will use to log on to VIRTEL (the suggested value is APPLID=VIRTEL)
2. The TCP1= parameter must match the jobname of your z/OS TCP/IP stack (the suggested value TCPIP is usually correct)
3. If you prefer VIRTEL to display English language panels, then set LANG='E'
4. (e) Set the COUNTRY and DEFUTF8 parameters according to your country (see “Parameters of the VIRTCT”, page 48)
5. Set the COMPANY ADDR1 ADDR2 LICENCE EXPIRE CODE parameters using the license key supplied to you by Syspertec.

8. Run the job ASMTCT in the VIRTEL CNTL library to assemble VIRTCT01 into VIRT453.LOADLIB.

9. Edit member ARBOLOAD in the VIRTEL CNTL library:

1. Change LANG=EN to LANG=FR if French language is desired
2. Set LOAD= the name of your VIRTEL LOADLIB
3. Set SAMP= the name of your VIRTEL SAMPLIB
4. Set ARBO= the name of your VIRTEL ARBO file
5. Set VTAMLST= the name of a your VIRTEL CNTL library. The job will create a sample VTAMLST member in this library.
6. CHANGE ALL 'DBDCCICS' 'xxxxxx'  
where xxxxxx is the APPLID of your CICS system.
7. If you changed the APPLID of VIRTEL in step 4 from its default value VIRTEL, then you must also change the ACBNAME= parameter in step VTAMDEF near the end of the ARBOLOAD job. The value of ACBNAME= in ARBOLOAD must match the value of APPLID= in VIRTCT01.

Submit the ARBOLOAD job. This creates your VIRTEL configuration (the ARBO file) and a sample VTAMLST member VIRTAPPL.

Note: If you need to rerun the ARBOLOAD job, you must change PARM='LOAD,NOREPL' to PARM='LOAD,REPL'

If you wish to completely start over from the beginning, you can run the job ARBOBASE to delete and reinitialize the ARBO file, followed by a rerun of the ARBOLOAD job.

10. Submit the job ASMMOD from the VIRTEL CNTL library. This job assembles the VIRTEL logon mode table (MODVIRT) into your SYS1.VTAMLIB dataset. You will need to set the QUAL= parameter to match the high-level qualifiers of your SAMPLIB dataset.

11. Copy the VIRTAPPL member (created by the ARBOLOAD job in step 8) from the VIRTEL CNTL library into your SYS1.VTAMLST dataset. Now activate the VTAMLST member using this command:

V NET,ACT,ID=VIRTAPPL

12. Edit the procedure VIRTEL4 in your VIRTEL CNTL library so that the high-level qualifiers match the names you used when you loaded the files in step 4. Copy the procedure to your system PROCLIB, renaming it as VIRTEL.

13. Ask your security administrator to create a userid for the VIRTEL started task, and to authorize this userid to access the datasets you created in step 3. This userid must also have an OMVS segment which authorizes VIRTEL to use TCP/IP. Your security administrator can use the job RACFSTC in the VIRTEL SAMPLIB as an example.

14. Start VIRTEL

15. You can now logon to VIRTEL from a 3270 terminal using the APPLID specified in the VIRTCT01, and you can display the VIRTEL Web Access menu in your web browser using URL <http://n.n.n.n:41001> where n.n.n.n is the IP address of your z/OS system.



16. Apply any "update" maintenance from the file virtel453updtnnnn.zip file according the instructions in theReadme-updtnnnn.txt file included in the zip file. If the zip file does not exist, skip this step.
17. The supplied system is configured with security disabled. If you wish, you can activate external security using RACF, ACF2, or TOP SECRET; please refer to the "[security chapter](#)", page 98.

# 4. Installing VIRTEL under VSE

## 4.1. Installation Procedure

Installation of VIRTEL under VSE consists of the following steps. Each step is described in detail in the sections which follow.

- Load the installation jobs into the POWER READER QUEUE
- Define the VIRT453.SUBLIB sublibrary
- Load the CIL and SSL libraries
- Define the files VIRARBO, VIRSWAP and VIRSTAT
- Define the files VIRCMP3, VIRCAPT and SAMPTRF
- Define the files HTMLTRF and VIRHTML
- Assemble the VIRTCT
- Assemble the VTAM mode table
- Update the VIRARBO file (ARBOLOAD)
- Define the VTAM application relays
- Define the VIRTEL start procedure

### 4.1.1. Loading the installation jobs

The installation jobs are delivered on an unlabeled 3480 tape cartridge. To load the installation jobs into the POWER reader queue, enter the command `S RDR, cuu` at the VSE console (where `cuu` represents the address of the tape drive on which you have mounted the cartridge). The following jobs will be loaded into your Reader Queue with `DISP=L, CLASS=0`:

#### **VIRTLIB**

| define the VIRT453.SUBLIB sublibrary

#### **VIRTCIL**

| load executable modules into the CIL

#### **VIRTSSL**

| load source modules into the SSL

#### **VIRAPI**

| load the VIRAPI macro library

**VIRFA29**

- | load the FA29 macro library

**VIRSAPI**

- | load the SCRNAPI macro library

**VIRTVS**

- | define the VSAM files, comprising the following steps:

**VIRTVS1**

- | | define VIRARBO and VIRSWAP files

**VIRTVS2**

- | | initialise VIRARBO file

**VIRTVS3**

- | | define VIRSTAT file

**VIRTVS4**

- | | define VIRCMP3 file

**VIRTVS5**

- | | define VIRCAPT file

**VIRTVS6**

- | | define SAMPTRF file

**VIRTVS7**

- | | define HTMLTRF file

**VIRTVS8**

- | | load SAMPTRF file

**VIRTVS9**

- | | define VIRHTML file

**VIRTCT**

- | VIRTEL parameter table assembly example

**VIRCONF**

- | VIRARBO batch update (ARBOLOAD)

**VIRMOD**

- | VTAM mode table assembly

**VIRTAPPL**

- | VTAM application major node example

**VIRGROUP**

- | CICS resource definition example

**VIRTEL**

- | VIRTEL execution JCL example

**Note:** You will need to modify certain of the installation jobs before submitting them. Once the jobs have been read onto the POWER queue, you can copy them to an ICCF library (using ICCF option 3224 Operations - Manage Batch Queues – Input Queue – Copy to Primary Library) or read them into your VM machine for editing.

#### 4.1.1.1. Sites installing VIRTEL for the first time

Jobs VIRTLIB, VIRTCL, VIRTSSL, VIRTVS, VIRTCT, VIRMOD, and VIRTAPPL must be executed as described below.

#### 4.1.1.2. Sites upgrading from a previous version

Execute jobs VIRTLIB, VIRTCL and VIRTSSL to create a new VIRT453.SUBLIB. Change your VIRTEL execution JCL to reference the new sublibrary. You can retain your existing VSAM files.

#### 4.1.1.3. Sites using VIRTEL Web Access

The files required for VIRTEL Web Access base functions are loaded in steps VIRTVS6, VIRTVS7, VIRTVS8, and VIRTVS9 of job VIRTVS. If you wish to use VIRTEL Host-Web Services to script your 3270 applications, run job VIRSAPI also.

#### 4.1.1.4. Sites using VIRTEL A2A

Customers wishing to use VIRTEL Application-to-Application functions should also run jobs VIRFA29 and VIRAPI.

### 4.1.2. Defining the library

```
* $$ JOB JNM=VIRTLIB,CLASS=0,DISP=L
* $$ LST DA
// JOB VIRTLIB
* *****
* * VIRTLIB *          CREATE VIRT453 LIBRARY *
* *****
* *
* * THIS JOB IS SUPPLIED AS AN EXAMPLE ONLY AND MUST BE MODIFIED *
* * BEFORE EXECUTION *
* *
* *****
// EXEC IDCAMS,SIZE=AUTO
DELETE (VSE.VIRT453.LIBRARY ) -
      CLUSTER -
      PURGE -
      CATALOG (VSESP.USER.CATALOG )
SET MAXCC=0
DEFINE CLUSTER ( -
      NAME (VSE.VIRT453.LIBRARY ) -
      TRACKS (150 25) -
      SHAREOPTIONS (3) -
      RECORDFORMAT (NOCIFORMAT) -
      VOLUMES (SYSWK1) -
      NOREUSE -
      NONINDEXED -
      TO (99366)) -
      DATA (NAME (VSE.VIRT453.LIBRARY.DATA ) ) -
      CATALOG (VSESP.USER.CATALOG )
IF LASTCC NE 0 THEN CANCEL JOB
/*
// OPTION STDLABEL=ADD
// DLBL VIRT453, 'VSE.VIRT453.LIBRARY' , , VSAM, CAT=VSESPUC
/*
// EXEC IESVCLUP,SIZE=AUTO
A VSE.VIRT453.LIBRARY VIRT453 VSESPUC OLD KEEP
/*
// EXEC LIBR,PARM='MSHP'
      DEFINE LIB=VIRT453 REPLACE=YES
      DEFINE SUBLIB=VIRT453.SUBLIB REPLACE=YES
```

```

/*
/&
* $$ E0J

```

*VIRTLIB : JCL to define the sublibrary (VSE)*

Job VIRTLIB contains an example of JCL to define the library which will contain the VIRTEL executable modules and source books. This job is provided as an example, and may need to be modified prior to execution. The name VIRTnnn.SUBLIB indicates the VIRTEL version, for example VIRT453.SUBLIB for version 4.53. Parameters VOLUMES(SYSWK1), and possibly the cluster name and catalog name, may need to be modified.

#### 4.1.3. Loading the executable modules

```

* $$ JOB JNM=VIRTCIL,CLASS=0,DISP=L
* $$ LST DA
// JOB VIRTCIL
* *****
* * VIRTCIL *          CATALOG PROGRAM PHASES IN CORE IMAGE LIBRARY *
* *****
* *
* * AT THE PAUSE, ENTER YOUR DLBL AND LIBDEF FOR THE CIL SUBLIB *
* *
* * // DLBL VIRT453, 'VSE.VIRT453.LIBRARY', ,VSAM,CAT=VSESPUC *
* * // LIBDEF PHASE,CATALOG=VIRT453.SUBLIB *
* *
* *****
// PAUSE ENTER YOUR LIBDEF PHASE STATEMENT AS IN THE ABOVE EXAMPLE
// OPTION CATAL
INCLUDE

(object modules)

/*
// EXEC LNKEDT,SIZE=512K
/&
* $$ E0J

```

*VIRTCIL : JCL to load the executable modules (VSE)*

Start the job to load the executable modules by entering the POWER command R RDR,VIRTCIL

When this job executes, a // PAUSE card will ask you to enter a LIBDEF statement to specify the name of the library into which the modules are to be loaded. Enter // LIBDEF PHASE,CATALOG=xxxxx where xxxxx represents the name of the sublibrary you defined in the previous job.

#### 4.1.4. Loading the source modules

```

* $$ JOB JNM=VIRTSSL,CLASS=0,DISP=L
* $$ LST DA
// JOB VIRTSSL
* *****
* * VIRTSSL *          CATALOG SOURCE BOOKS IN SSL *
* *****
* *
* * AT THE PAUSE, ENTER THE NAME OF THE SUB-LIBRARY *
* * FOR CATALOGING THE VIRTEL SOURCE BOOKS *
* *
* * EXAMPLE: // SETPARM SUB='VIRT453.SUBLIB' *
* *
* *****
// PAUSE ENTER YOUR SETPARM CARD AS SHOWN ABOVE
// EXEC PGM=LIBR,PARM=' ACCESS SUBLIB=&SUB'

```

*(source books)*

```
/*
/ &
* $$ E0J
```

*VIRTSSL : JCL to load the source modules (VSE)*

Start the job to load the source modules by entering the POWER command R RDR,VIRTSSL followed by R RDR,VIRFA29 then R RDR,VIRAPI and R RDR,VIRSAPI as necessary.

When these jobs execute, a // PAUSE card will ask you to enter a SETPARM statement specifying the name of the library into which the modules are to be loaded. Enter // SETPARM SUB='xxxxxxx' where xxxxxx represents the name of the sublibrary you defined in the first job.

```
* $$ JOB JNM=VIRFA29,CLASS=0,DISP=L
* $$ LST DA
// JOB VIRFA29
* *****
* * VIRFA29 *          CATALOG SOURCE BOOKS FOR FA29 API          *
* *****
* *
* * AT THE PAUSE, ENTER THE NAME OF THE SUB-LIBRARY                *
* * FOR THE FA29 MACRO SOURCE BOOKS                                *
* *
* *          EXAMPLE: // SETPARM SUB='VIRT453.SUBLIB'                *
* *
* *****
// PAUSE ENTER YOUR SETPARM CARD AS SHOWN ABOVE
// EXEC PGM=LIBR,PARM=' ACCESS SUBLIB=&SUB'
```

*(FA29API source books)*

```
/*
/ &
* $$ E0J
```

*VIRFA29 : JCL to load the FA29 macros (VSE)*

```
* $$ JOB JNM=VIRAPI,CLASS=0,DISP=L
* $$ LST DA
// JOB VIRAPI
* *****
* * VIRAPI *          CATALOG SOURCE BOOKS FOR VIRAPI          *
* *****
* *
* * AT THE PAUSE, ENTER THE NAME OF THE SUB-LIBRARY                *
* * FOR THE VIRAPI MACRO SOURCE BOOKS                                *
* *
* *          EXAMPLE: // SETPARM SUB='VIRT453.SUBLIB'                *
* *
* *****
// PAUSE ENTER YOUR SETPARM CARD AS SHOWN ABOVE
// EXEC PGM=LIBR,PARM=' ACCESS SUBLIB=&SUB'
```

*(VIRAPI source books)*

```
/*
/ &
* $$ E0J
```

*VIRAPI : JCL to load the VIRAPI macros (VSE)*

```
* $$ JOB JNM=VIRSAPI,CLASS=0,DISP=L
* $$ LST DA
```

```
// JOB VIRSAPI
* *****
* * VIRSAPI *      CATALOG SOURCE BOOKS FOR SCRNAPI *
* *****
* *
* * AT THE PAUSE, ENTER THE NAME OF THE SUB-LIBRARY *
* * FOR THE SCRNAPI MACRO SOURCE BOOKS *
* *
* *      EXAMPLE:  // SETPARM SUB='VIRT453.SUBLIB' *
* *
* *****
// PAUSE ENTER YOUR SETPARM CARD AS SHOWN ABOVE
// EXEC PGM=LIBR,PARM=' ACCESS SUBLIB=&SUB'

(SCRNAPI source books)

/*
/&
* $$ E0J
```

VIRSAPI : JCL to load the SCRNAPI macros (VSE)

#### 4.1.5. Defining the VIRARBO and VIRSWAP files

```
// JOB VIRTVS
// SETPARM TAPE=590
* *****
* * AT THE PAUSE, ENTER THE UNIT ADDRESS OF THE TAPE DRIVE *
* * FOR THE VIRTEL INSTALLATION TAPE *
* *
* *      EXAMPLE:  // SETPARM TAPE=590 *
* *
* *****
// PAUSE ENTER YOUR SETPARM CARD AS SHOWN ABOVE
* *****
* * VIRTVS1 *      DEFINITION OF VIRARBO AND VIRSWAP FILES *
* *****
// DLBL IJSYSUC, 'VSESP.USER.CATALOG', ,VSAM
// EXEC IDCAMS,SIZE=AUTO
DELETE (VIRTEL.ARBO ) -
    CLUSTER -
    PURGE -
    CATALOG (VSESP.USER.CATALOG )
SET MAXCC=0
DEFINE CLUSTER ( -
    NAME (VIRTEL.ARBO ) -
    RECORDS(500 100) -
    SHAREOPTIONS (4 3) -
    RECSZ (600 4089) -
    VOLUMES (SYSWK1) -
    KEYS (9 0) -
    TO (99366))-
DATA (NAME (VIRTEL.ARBO.DATA )) -
INDEX (NAME (VIRTEL.ARBO.INDEX )) -
    CATALOG (VSESP.USER.CATALOG )
IF LASTCC NE 0 THEN CANCEL JOB
DELETE (VIRTEL.SWAP ) -
    CLUSTER -
    PURGE -
    CATALOG (VSESP.USER.CATALOG )
SET MAXCC=0
DEFINE CLUSTER ( -
    NAME (VIRTEL.SWAP ) -
    RECORDS(200 50) -
    SHAREOPTIONS (2 3) -
    RECSZ (600 4089) -
    VOLUMES (SYSWK1) -
```

```

        REUSE -
        KEYS (16 0) -
        TO (99366))-
DATA  (NAME (VIRTEL.SWAP.DATA          )) -
INDEX (NAME (VIRTEL.SWAP.INDEX        )) -
        CATALOG (VSESP.USER.CATALOG    )
IF LASTCC NE 0 THEN CANCEL JOB
/*

```

*VIRTVS1 : JCL to define the VIRARBO and VIRSWAP files (VSE)*

Step VIRTVS1 of job VIRTVS contains an example of defining the VIRARBO and VIRSWAP files. This job is provided as an example, and may need to be modified prior to execution. The parameters SETPARM TAPE=590 and VOLUMES(SYSWK1), and possible the catalog name, may need to be modified.

#### 4.1.6. Initialisation of the VIRARBO file

```

* *****
* * VIRTVS2 *          INITIALISATION OF VIRARBO FILE          *
* *****
// DLBL VIRARBO, 'VIRTEL.ARBO', , VSAM, CAT=VSESPUC
// PAUSE **** VIRTEL **** MOUNT INSTALLATION TAPE
// ASSGN SYS004, &TAPE
// MTC REW, SYS004
// MTC FSF, SYS004, 2          1=FRANCAIS, 2=ANGLAIS
// EXEC IDCAMS, SIZE=AUTO
      REPRO IFILE(BANDE ENV(PDEV(2400) NOLABEL RECFM(VB) BLKSZ(32000))) -
          OFILE(VIRARBO)
/*

```

*VIRTVS2 : JCL to initialise the VIRARBO file (VSE)*

Step VIRTVS2 of job VIRTVS loads the base configuration definitions into the VIRARBO file. The default language is English. To load the French language version of the base configuration, change the // MTC FSF, SYS004, 2 card to // MTC FSF, SYS004, 1 before submitting this job.



#### 4.1.7. Defining the VIRSTAT file

```

* *****
* * VIRTVS3 *      DEFINITION OF VIRSTAT FILE      *
* *****
// DLBL IJSYSUC, 'VSESP.USER.CATALOG', ,VSAM
// EXEC IDCAMS,SIZE=AUTO
DELETE (VIRTEL.STAT ) -
    CLUSTER -
    PURGE -
    CATALOG (VSESP.USER.CATALOG )
SET MAXCC=0
DEFINE CLUSTER ( -
    NAME (VIRTEL.STAT ) -
    RECORDS (500 100)-
    SHAREOPTIONS (2) -
    RECSZ (124 620) -
    RECORDFORMAT (FIXBLK (124 ))-
    VOLUMES (SYSWK1) -
    NOREUSE -
    NONINDEXED -
    FREESPACE (15 7) -
    TO (99366))-
DATA (NAME (VIRTEL.STAT.DATA ) ) -
    CATALOG (VSESP.USER.CATALOG )
IF LASTCC NE 0 THEN CANCEL JOB
/*

```

*VIRTVS3 : JCL to define the VIRSTAT file (VSE)*

Step VIRTVS3 of job VIRTVS contains an example of defining the VIRSTAT file. This job is provided as an example, and may need to be modified prior to execution. The VIRSTAT file is required unless the STATS parameter of the VIRTCT is set to NO.

#### 4.1.8. Defining the VIRCMP3 file

```

* *****
* * VIRTVS4 *      DEFINITION AND INITIALIZATION OF VIRCMP3 FILE *
* *****
// DLBL IJSYSUC, 'VSESP.USER.CATALOG', ,VSAM
// EXEC IDCAMS,SIZE=AUTO
DELETE (VIRTEL.CMP3 ) -
    CLUSTER -
    PURGE -
    CATALOG (VSESP.USER.CATALOG )
SET MAXCC=0
DEFINE CLUSTER ( -
    NAME (VIRTEL.CMP3 ) -
    RECORDS(200 50)-
    SHAREOPTIONS (2 3) -
    RECSZ (600 8185) -
    VOLUMES (SYSWK1) -
    KEYS (9 0) -
    TO (99366))-
DATA (NAME (VIRTEL.CMP3.DATA ) ) -
INDEX (NAME (VIRTEL.CMP3.INDEX ) ) -
    CATALOG (VSESP.USER.CATALOG )
IF LASTCC NE 0 THEN CANCEL JOB
/*
// DLBL VIRCMP3, 'VIRTEL.CMP3', 2099/365,VSAM,CAT=VSESPUC
// EXEC IESVSMLD,SIZE=AUTO          LOAD DUMMY RECORD INTO VIRCMP3
80,K,VIRCMP3
ZZZ
/*

```

*VIRTVS4 : JCL to define the VIRCMP3 file (VSE)*

Step VIRTVS4 of job VIRTVS contains an example of defining the VIRCMP3 file. This job is provided as an example, and may need to be modified prior to execution. The VIRCMP3 file is used by the level 3 compression feature of VIRTEL/PC, and is required unless the COMPR3 parameter of the VIRTCT is set to NO.

#### 4.1.9. Defining the VIRCAPT file

```
* *****
* * VIRTVS5 *      DEFINITION AND INITIALIZATION OF VIRCAPT FILE *
* *****
// DLBL IJSYSUC, 'VSESP.USER.CATALOG', , VSAM
// EXEC IDCAMS, SIZE=AUTO
DELETE (VIRTEL.CAPT
        CLUSTER -
        PURGE -
        CATALOG (VSESP.USER.CATALOG
SET MAXCC=0
DEFINE CLUSTER ( -
        NAME (VIRTEL.CAPT
        RECORDS(200 50) -
        SHAREOPTIONS (2 3) -
        RECSZ (600 8185) -
        VOLUMES (SYSWK1) -
        KEYS (16 0) -
        TO (99366)) -
DATA (NAME (VIRTEL.CAPT.DATA
INDEX (NAME (VIRTEL.CAPT.INDEX
        CATALOG (VSESP.USER.CATALOG
IF LASTCC NE 0 THEN CANCEL JOB
/*
// DLBL VIRCAPT, 'VIRTEL.CAPT', 2099/365, VSAM, CAT=VSESPUC
// EXEC IESVSM, SIZE=AUTO          LOAD DUMMY RECORD INTO VIRCAPT
80,K,VIRCAPT
ZZZ
/*
```

*VIRTVS5 : JCL to define the VIRCAPT file (VSE)*

Step VIRTVS5 of job VIRTVS contains an example of defining the VIRCAPT file. This job is provided as an example, and may need to be modified prior to execution. The VIRCAPT file is used by the videotext page capture feature, and is referenced by the FCAPT parameter of the VIRTCT.

#### 4.1.10. Defining the SAMPTRF file

```
* *****
* * VIRTVS6 *      DEFINITION AND INITIALIZATION OF SAMPTRF FILE *
* *****
// DLBL IJSYSUC, 'VSESP.USER.CATALOG', , VSAM
// EXEC IDCAMS, SIZE=AUTO
DELETE (VIRTEL.SAMP.TRSF
        CLUSTER -
        PURGE -
        CATALOG (VSESP.USER.CATALOG
SET MAXCC=0
DEFINE CLUSTER ( -
        NAME(VIRTEL.SAMP.TRSF
        TO (99365) -
        FREESPACE (0 50) -
        SHAREOPTIONS (2) -
        INDEXED -
        KEYS (16 0) -
        RECORDSIZE (100 32758) -
        USECLASS (0) -
        VOLUMES (SYSWK1)) -
DATA (NAME(VIRTEL.SAMP.TRSF.DATA
```

```

        SPANNED -
        TRACKS(75 15) -
        CISZ (4096)) -
INDEX (NAME(VIRTEL.SAMP.TRSF.INDEX          ) -
        TRACKS(5 1) -
        CISZ (512)) -
        CATALOG (VSESP.USER.CATALOG          )
/*
// DLBL INWFILE, 'VIRTEL.SAMP.TRSF', 2099/365, VSAM, CAT=VSESPUC
// EXEC IESVSMLD, SIZE=AUTO          LOAD DUMMY RECORD INTO INWFILE
80,K, INWFILE
$$$$IWS.WORKREC.INW$TEMP
/*

```

*VIRTVS6 : JCL to define the SAMPTRF file (VSE)*

Step VIRTVS6 of job VIRTVS contains an example of defining the SAMPTRF file. This job is provided as an example, and may need to be modified prior to execution. The SAMPTRF file contains sample HTML page templates and other elements for the VIRTEL Web Access feature, and is referenced by the UFILEX parameter of the VIRTCT.

#### 4.1.11. Defining the HTMLTRF file

```

* *****
* * VIRTVS7 *      DEFINITION AND INITIALIZATION OF HTMLTRF FILE *
* *****
// DLBL IJSYSUC, 'VSESP.USER.CATALOG', , VSAM
// EXEC IDCAMS, SIZE=AUTO
DELETE (VIRTEL.HTML.TRSF          ) -
        CLUSTER -
        PURGE -
        CATALOG (VSESP.USER.CATALOG          )
SET MAXCC=0
DEFINE CLUSTER ( -
        NAME(VIRTEL.HTML.TRSF          ) -
        RECORDS (2500 1000) -
        TO (99365) -
        FREESPACE (0 50) -
        SHAREOPTIONS (2) -
        INDEXED -
        KEYS (16 0) -
        RECORDSIZE (100 32758) -
        USECLASS (0) -
        VOLUMES (SYSWK1)) -
DATA (NAME(VIRTEL.HTML.TRSF.DATA          ) -
        SPANNED -
        TRACKS(75 15) -
        CISZ (4096)) -
INDEX (NAME(VIRTEL.HTML.TRSF.INDEX          ) -
        TRACKS(5 1) -
        CISZ (512)) -
        CATALOG (VSESP.USER.CATALOG          )
/*
// DLBL HTMLTRF, 'VIRTEL.HTML.TRSF', 2099/365, VSAM, CAT=VSESPUC
// EXEC IESVSMLD, SIZE=AUTO          LOAD DUMMY RECORD INTO HTMLTRF
80,K, HTMLTRF
$$$$IWS.WORKREC.INW$TEMP
/*

```

*VIRTVS7 : JCL to define the HTMLTRF file (VSE)*

Step VIRTVS7 of job VIRTVS contains an example of defining the HTMLTRF file. This job is provided as an example, and may need to be modified prior to execution. The HTMLTRF file is used by the VIRTEL Web Access feature to store HTML pages, and is referenced by the UFILEX parameter of the VIRTCT.

#### 4.1.12. Loading the SAMPTRF file

```

* *****
* * VIRTVS8 *          LOAD DATA INTO SAMPTRF FILE          *
* *****
// DLBL SAMPTRF, 'VIRTEL.SAMP.TRSF', ,VSAM,CAT=VSESPUC
// PAUSE **** VIRTEL **** MONTEZ LA BANDE D'INSTALLATION
// ASSGN SYS004,&TAPE
// MTC REW,SYS004
// MTC FSF,SYS004,3
// EXEC IDCAMS,SIZE=AUTO
    REPRO IFILE(BANDE ENV(PDEV(2400) NOLABEL RECFM(VB) BLKSZ(32000))) -
        OFFILE(SAMPTRF) REPLACE
/*

```

*VIRTVS8 : JCL to load the SAMPTRF file (VSE)*

Step VIRTVS8 of job VIRTVS contains an example of the JCL required to load the sample HTML pages into the SAMPTRF file. This job is required for sites using VIRTEL Web Access.

#### 4.1.13. Defining the VIRHTML file

```

* *****
* * VIRTVS9 *          DEFINITION AND INITIALIZATION OF VIRHTML FILE *
* *****
// DLBL IJSYSUC, 'VSESP.USER.CATALOG', ,VSAM
// EXEC IDCAMS,SIZE=AUTO
    DELETE (VIRTEL.HTML
        CLUSTER -
        PURGE -
        CATALOG (VSESP.USER.CATALOG
        SET MAXCC=0
    DEFINE CLUSTER ( -
        NAME(VIRTEL.HTML
        RECORDS (50 100) -
        TO (99365) -
        FREESPACE (0 50) -
        SHAREOPTIONS (2) -
        INDEXED -
        KEYS (64 0) -
        RECORDSIZE (100 32758) -
        USECLASS (0) -
        VOLUMES (SYSWK1)) -
    DATA (NAME(VIRTEL.HTML.DATA
        SPANNED -
        CISZ (4096)) -
    INDEX (NAME(VIRTEL.HTML.INDEX
        CISZ (512)) -
        CATALOG (VSESP.USER.CATALOG
/*
// DLBL VIRHTML, 'VIRTEL.HTML', 2099/365,VSAM,CAT=VSESPUC
// EXEC IESVSMLD,SIZE=AUTO          LOAD DUMMY RECORD INTO VIRHTML
80,K,VIRHTML
ZZZ
/*

```

*VIRTVS9 : JCL to define the VIRHTML file (VSE)*

Step VIRTVS9 of job VIRTVS contains an example of defining the VIRHTML file. This job is provided as an example, and may need to be modified prior to execution. The VIRHTML file is used by the VIRTEL Web Access feature to store the names of E-mail correspondents, and is referenced by the HTVSAM parameter of the VIRTCT.

#### 4.1.14. Assembling the VIRTCT

Job VIRTCTUS contains an example of assembling the VIRTEL parameter table (the VIRTCT). Since the VIRTCT parameters are common across the VSE, MVS and VM environments, please refer to section VIRTCT 47.

Users in France should use job VIRTCTFR instead of VIRTCTUS

#### 4.1.15. Assembling the MODVIRT mode table

```
* $$ JOB JNM=VIRMOD,CLASS=0,DISP=L
* $$ LST DA
// JOB VIRMOD
* *****
* * VIRMOD *          ASSEMBLY OF THE VTAM MODE TABLE *
* *****
* *
* * THIS JOB IS SUPPLIED AS AN EXAMPLE ONLY AND MUST BE MODIFIED *
* * BEFORE EXECUTION *
* *
* *****
// DLBL VIRT453, 'VSE.VIRT453.LIBRARY', ,VSAM,CAT=VSESPUC
// LIBDEF PHASE,CATALOG=PRD2.CONFIG
// LIBDEF SOURCE,SEARCH=(VIRT453.SUBLIB,PRD1.BASE)
// OPTION CATAL
    PHASE MODVIRT,*
// EXEC ASSEMBLY,SIZE=512K
    COPY MODVIRT
/*
// EXEC LNKEDT,SIZE=512K
/*
/&
* $$ E0J
```

*VIRMOD : Assembling the MODVIRT mode table (VSE)*

Job VIRMOD contains an example of the JCL required to assemble the VTAM mode table (MODVIRT) supplied with VIRTEL.

#### 4.1.16. Updating the VIRARBO file (ARBOLOAD)

```
* $$ JOB JNM=VIRCONF,CLASS=0,DISP=L
* $$ LST DA
// JOB VIRCONF
* *****
* * VIRCONF *          LOAD CONFIGURATION DATA (ARBOLOAD) *
* *****
* *
* * THIS JOB IS SUPPLIED AS AN EXAMPLE ONLY AND MUST BE MODIFIED *
* * BEFORE EXECUTION *
* *
* *****
// LIBDEF *,SEARCH=(VIRT453.SUBLIB)
// DLBL VIRARBO, 'VIRTEL.ARB0', ,VSAM,CAT=VSESPUC
// SETPARM LANG=EN
// SETPARM WEB=YES
// SETPARM VMACROS=NO
// SETPARM SMTP=NO
// SETPARM IMSW=NO
// SETPARM VHOST=NO
// SETPARM PLUG=NO
// SETPARM VSR=NO
// SETPARM IPAD=NO
// SETPARM MINITEL=NO
```

```
// SETPARM PCMGMT=NO
// SETPARM NTTCP=NO
// SETPARM XOT=NO
// SETPARM NPSIFC=NO
// SETPARM NPSIGAT=NO
// SETPARM ANTIFC=NO
// SETPARM CFTGATE=NO
// SETPARM CFTPCNE=NO
// SETPARM MULTSES=NO
// SETPARM VIRSECU=NO
// IF WEB NE YES THEN
// GOTO WEB
// EXEC VIRCONF,PARM='LOAD,LANG=&LANG'

(configuration statements for VIRTEL Web Access feature)

/*
/. WEB
// IF XOT NE YES THEN
// GOTO XOT
// EXEC VIRCONF,PARM='LOAD,LANG=&LANG'

(configuration statements for XOT feature)

/*
/. XOT

(etc)

/&
* $$ E0J
```

**VIRCONF** : ARBOLOAD job to update the VIRARBO file (VSE)

Job VIRCONF contains an example of a job to load configuration elements into the VIRARBO file. This is the equivalent of the MVS job known as ARBOLOAD. Before running this job, you will need to make the following modifications:

- Select the desired features (for example, WEB=YES, XOT=YES)
- Change all 'DBDCCICS' to the APPLID of your CICS system

Users in France may also change LANG=EN to LANG=FR to generate French language versions of the configuration elements

#### 4.1.17. Cataloging the VTAM application book

```
* $$ JOB JNM=VIRTAPPL,CLASS=0,DISP=L
* $$ LST DA
// JOB VIRTAPPL
* *****
* * VIRTAPPL *          EXAMPLE OF APPLICATION MAJOR NODE FOR VIRTEL *
* *****
* *
* * THIS JOB IS SUPPLIED AS AN EXAMPLE ONLY AND MUST BE MODIFIED *
* * BEFORE EXECUTION *
* *
* *****
// EXEC LIBR
ACCESS SUBLIB=PRD2.CONFIG
CATALOG VIRTAPPL.B          REPLACE=YES
* ----- *
* Product      : Virtel *
* Description  : Main ACB for VIRTEL application *
* ----- *
VIRTEL  APPL AUTH=(PASS,ACQ,SP0)

(APPL statements for other VIRTEL relays)
```

```

/+
/*
/&
* $$ E0J

```

*VIRTAPPL : Cataloging the application major node (VSE)*

Job VIRTAPPL contains an example of cataloging the VTAM application book. The VTAM application node VIRTAPPL must be activated before starting VIRTEL. This job is provided as an example, and may need to be modified prior to execution.

#### 4.1.18. Defining the CICS resources

```

* $$ JOB JNM=VIRGROUP,CLASS=A,DISP=D,NTFY=YES
* $$ LST DA
// JOB VIRGROUP CREATION CICS CSD GROUP VIRTEL
* *****
* * VIRGROUP *          CICS RESOURCE DEFINITIONS FOR VIRTEL *
* *****
* *
* * THIS JOB IS SUPPLIED AS AN EXAMPLE ONLY AND MUST BE MODIFIED *
* * BEFORE EXECUTION *
* *
* *****
* *
* * SEE IJSYSRS.SYSLIB/STDLABUP.PROC FOR DEFAULT DLBL DFHCSD *
* * // DLBL DFHCSD,'CICS.CSD',0,VSAM,CAT=VSESPUC *
* *
* *****
// EXEC DFHCSDUP,SIZE=AUTO
* VIRTEL 3270 TERMINALS FOR WEB2HOST
DEFINE  TE(T000) G(VIRTEL) TY(VSELU2Q) NE(RHTVT000) PRINTER(I000)
        DESC(VIRTEL WEB TO HOST TERMINAL)
DEFINE  TE(T001) G(VIRTEL) TY(VSELU2Q) NE(RHTVT001) PRINTER(I001)
        DESC(VIRTEL WEB TO HOST TERMINAL)
DEFINE  TE(T002) G(VIRTEL) TY(VSELU2Q) NE(RHTVT002) PRINTER(I002)
        DESC(VIRTEL WEB TO HOST TERMINAL)
etc.
* VIRTEL 3284 PRINTERS FOR WEB2HOST
DEFINE  TE(I000) G(VIRTEL) TY(VSELU3Q) NE(RHTIM000)
        DESC(VIRTEL WEB TO HOST PRINTER)
DEFINE  TE(I001) G(VIRTEL) TY(VSELU3Q) NE(RHTIM001)
        DESC(VIRTEL WEB TO HOST PRINTER)
DEFINE  TE(I002) G(VIRTEL) TY(VSELU3Q) NE(RHTIM002)
        DESC(VIRTEL WEB TO HOST PRINTER)
etc.
* ADD VIRTEL GROUP TO STARTUP LIST
ADD     GROUP(VIRTEL) LIST(VSELIST)
/*
/&
* $$ E0J

```

*VIRGROUP : Defining the CICS resources (VSE)*

Job VIRGROUP contains an example of defining the the CICS resources which are correspond to the relays and virtual printers used by VIRTEL Web Access. This job is provided as an example, and may need to be modified prior to execution.

## 4.2. Applying VWA User Interface Update

Under certain circumstances it may be necessary to apply maintenance in the form of User Interface Updates. These may be distributed either by e-mail, or available on Syspertec FTP Server.

An update is available as a ZIP file containing the cumulative days update the version. The file is represented in the form VirtelxxxUpdtnnnn.zip where xxx is the version of Virtel to which it relates and nnnn the reference of the update itself. Once unzipped, the file content is in the form of a tree where each folder contains one or more files grouped by category, the root contains a file named updtnnnn.txt which summarized the history of changes and any special instructions to operate.

Generally, the file still contains a sub directory named " W2H " whose content must be reloaded into the W2H-DIR using one of the methods described in section "Uploading HTML Pages" from document "Virtel Web Access User Guide".

If you are installing a Virtel version 453, you must:

- First, install the update number virtel453updt5228.zip
- Then install that last level of update available, even if the version number is greater than 453.

## 4.3. Executing VIRTEL In A VSE Environment

Job VIRTEL contains an example of the VSE startup JCL for VIRTEL. Program VIR0000 reads a parameter card indicating the suffix of the VIRTCT to be used. This suffix must be two characters long and must start in column 1 of the parameter card. In the example supplied, the suffix is 01, indicating that parameter table VIRTCT01 is to be used. The TCT suffix may optionally be followed by a comma and the VTAM APPLID. If the APPLID is not specified then the value in the VIRTCT is used.

The partition used must have a size of at least 1.5MB and must have 1MB of GETVIS. The priority of the VIRTEL partition must be immediately below that of VTAM.

```
* $$ JOB JNM=VIRTEL,CLASS=4,DISP=L,PRI=9
* $$ LST DA
// JOB VIRTEL
* *****
* * VIRTEL *          EXAMPLE JCL TO EXECUTE VIRTEL *
* *****
* *
* * THIS JOB IS SUPPLIED AS AN EXAMPLE ONLY AND MUST BE MODIFIED *
* * BEFORE EXECUTION *
* *
* *****
// OPTION SYSPARM='00'    MUST MATCH PARM ID=NN IN TCP/IP PARTITION
// LIBDEF *,SEARCH=(VIRT453.SUBLIB,PRD2.CONFIG,PRD1.BASE)
// DLBL VIRARBO,'VIRTEL.ARBO',,VSAM,CAT=VSESPUC
// DLBL VIRSWAP,'VIRTEL.SWAP',,VSAM,CAT=VSESPUC
// DLBL VIRCAPT,'VIRTEL.CAPT',,VSAM,CAT=VSESPUC
// DLBL VIRCMP3,'VIRTEL.CMP3',,VSAM,CAT=VSESPUC
// DLBL VIRHTML,'VIRTEL.HTML',,VSAM,CAT=VSESPUC
// DLBL SAMPTRF,'VIRTEL.SAMP.TRSF',,VSAM,CAT=VSESPUC
// DLBL HTMLTRF,'VIRTEL.HTML.TRSF',,VSAM,CAT=VSESPUC
// DLBL VIRSTAT,'VIRTEL.STAT',,VSAM,CAT=VSESPUC
* * OU BIEN // DLBL VIRSTAT,'VIRTEL.STAT',0,SD
* *          // EXTENT SYS001,SYSWK1,1,0,855,15
* *          // ASSGN SYS001,DISK,VOL=SYSWK1,SHR
// EXEC IESWAITT
// EXEC VIR0000,SIZE=40K,DSPACE=2M
01,VIRTEL
/*
// EXEC LISTLOG
```



```
/&
* $$ E0J
```

*VIRTEL startup JCL (VSE)*

#### 4.3.1. Specifying the TCP/IP partition

If you have more than one TCP/IP stack, you can use the `OPTION SYSPARM='nn'` statement to specify the ID of the TCP/IP stack. VIRTEL will attempt to connect to the TCP/IP partition which has `PARM='ID=nn'` in its JCL. If `OPTION` is not specified, VIRTEL will attempt to connect to the default TCP/IP whose ID is 00.

#### 4.3.2. Stopping VIRTEL

To stop VIRTEL, enter the command:

```
MSG xx,DATA=STOP
```

where `xx` is the identifier of the partition in which VIRTEL is running.

### 4.4. Applying PTFs

Under certain circumstances it may be necessary to apply maintenance in the form of PTFs. These may be distributed either by e-mail, or on a 3480 cartridge.

#### 4.4.1. Loading from a 3480 cartridge

PTFs are supplied on an unlabeled tape. The tape contains the PTFs to be applied at your site and has been generated by DITTO. The record length is 81 and the blocksize is 8100. To load the PTFs from the tape, enter the command `DITTO TC cuu` where `cuu` is the address of your tape unit. The contents of the tape will be placed in the VSE Punch. Copy the contents of the Punch into a library and add the necessary execution JCL before submitting the job.

#### 4.4.2. Application of the PTFs

To apply the PTFs, use the following JCL:

```
* $$ JOB JNM=PTFnxxx,CLASS=0,DISP=D,PRI=9
* $$ LST DA
// JOB PTFnnxxx
// EXEC MSHP
PATCH SUBLIB=VIRT453.SUBLIB
AFFECTS PHASE=modname
ALTER xxxx vvvvvvvv:rrrrrrrr
/*
/&
* $$ E0J
```

*JCL for applying PTFs (VSE)*

## 5. VTAM parameters

### 5.1. VTAM Parameters

This section describes the VTAM definitions required for VIRTEL. The same definitions are used in both the MVS and VSE environments.

### 5.2. Definition Of The VIRTEL Primary ACB

The primary ACB is defined by means of a VTAM APPL statement:

```
applname APPL AUTH=(PASS,ACQ,SP0)
```

#### **applnamere**

| presents the name of the ACB as it is defined in the APPLID statement of the VIRTCT.

An example of a VTAM application node is provided in member VIRTAPPL of the VIRTEL SAMPLIB dataset for MVS, or in the VIRTAPPL installation job for VSE.

### 5.3. Defining The VTAM Application Relays

Each terminal which logs on to a VTAM application via VIRTEL requires an application relay. An application relay is a VTAM LU, defined by means of a VTAM APPL card, which VIRTEL uses to represent the terminal when connecting to the VTAM application. These APPL cards are defined as follows:

```
relaynam APPL AUTH=(PASS,ACQ),MODETAB=tablenam,DLOGMOD=modename,EAS=1
```

#### **relaynam**

| Represents the name of the relay associated with the terminal. This name must match the name specified in the "Relay" field of the VIRTEL terminal definition.

#### **tablenam**

| Is the name of the logon mode table. For VIRTEL Web Access, use the standard IBM-supplied table ISTINCLM. For other types of relay, use the MODVIRT table supplied by VIRTEL.

**modename**

Is the name of the LOGMODE to be used for communication with the host application. For VIRTEL Web Access, use a standard IBM-supplied logmode such as SNX32702. For 3270 emulation via Minitel, use DLOGREL which is defined in the MODVIRT table supplied by VIRTEL.

**EAS=1**

Since each application relay only uses one session, specification of this parameter may reduce common area storage requirements.

## 5.4. MODETAB For Minitel, X25, And APPC

If you intend to use Minitel, X25, or APPC, then a mode table named MODVIRT must be assembled and link-edited into the library from which VTAM loads its mode tables. For MVS, a sample job is provided in the ASMMOD member of the VIRTEL SAMPLIB. For VSE, sample JCL is provided in the VIRMOD installation job.

The source for the MODVIRT mode table is defined as follows:

```
MODVIRT MODETAB
* LOGMODE for LUTYPE2 terminals
DLOGREL  MODEENT LOGMODE=DLOGREL,                                X
          FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',                X
          SECPROT=X'90',COMPROT=X'3080',RUSIZES=X'87F8',          X
          PSERVIC=X'028000000000185000007E00'
* LOGMODE for LUTYPE1 terminals
DLOGMINI MODEENT LOGMODE=DLOGMINI,                                X
          FMPROF=X'03',                                          X
          TSPROF=X'03',                                          X
          PRIPROT=X'B1',                                          X
          SECPROT=X'90',                                          X
          COMPROT=X'3040',          CONTENTION                    X
          RUSIZES=X'8589',          256-4096                      X
          PSERVIC=X'010000000000000000000000'
* LOGMODE for inversed GATE
DLOGANTI MODEENT LOGMODE=DLOGANTI,                                X
          FMPROF=X'03',TSPROF=X'03',PRIPROT=X'B1',SECPROT=X'90', X
          COMPROT=X'3040',RUSIZES=X'8989',                        X
          PSERVIC=X'010000000000000000000000'
* LOGMODE for inversed PCNE                                     @416
DLOGPCNE MODEENT LOGMODE=DLOGPCNE,                                @416X
          FMPROF=X'03',TSPROF=X'03',                              @416X
          PRIPROT=X'B0',SECPROT=X'B0',                            @416X
          COMPROT=X'0040',RUSIZES=X'8989',                        @416X
          PSERVIC=X'000000000000000000000000'                    @416
* LOGMODE for APPC lines (LU6.2)
LU62CONV MODEENT LOGMODE=LU62CONV,FMPROF=X'13',TSPROF=X'07',    X
          PRIPROT=X'B0',SECPROT=X'B0',COMPROT=X'D0B1',           X
          RUSIZES=X'8686',ENCR=B'0000',TYPE=0,                  X
          PSERVIC=X'060200000000000000000000300'
          MODEEND
          END
```

*VTAM logon mode table MODVIRT*

## 5.5. USSTAB For Minitels And PC's

For Minitel and VIRTEL/PC it may be necessary to provide a customized USS table in the VTAM library. An example USS table is shown in the figure below. A USS table is not necessary for VIRTEL Web Access access.

```

USSTAB
USSCMD  CMD=MA,REP=LOGON,FORMAT=BAL
USSPARM  PARM=APPLID,DEFAULT=VIRTEL2
USSCMD  CMD=P,REP=LOGON,FORMAT=BAL
USSPARM  PARM=APPLID,DEFAULT=VIRTEL2
USSPARM  PARM=DATA,DEFAULT='PC'
USSMSG   MSG=00,BUFFER=MSG00
USSMSG   MSG=01,BUFFER=MSG02
USSMSG   MSG=02,BUFFER=MSG02
USSMSG   MSG=03,BUFFER=MSG03
USSMSG   MSG=04,BUFFER=MSG04
USSMSG   MSG=05,BUFFER=MSG02
USSMSG   MSG=06,BUFFER=MSG02
USSMSG   MSG=07,BUFFER=MSG04
USSMSG   MSG=08,BUFFER=MSG02
USSMSG   MSG=09,BUFFER=MSG02
USSMSG   MSG=10,BUFFER=MSG10
USSMSG   MSG=11,BUFFER=MSG10
USSMSG   MSG=12,BUFFER=MSG02
MSG00    DC    Y(MSG0F-* -2)
          DC    X'0D',C'DEMANDE PRISE EN COMPTE',X'0D'
MSG0F    EQU    *
MSG02    DC    Y(MSG2F-* -2)
          DC    X'0D',C'CHOIX NON PREVU',X'0D'
MSG2F    EQU    *
MSG03    DC    Y(MSG3F-* -2)
          DC    X'0D',C'CODE RETOUR INCONNU',X'0D'
MSG3F    EQU    *
MSG04    DC    Y(MSG4F-* -2)
          DC    X'0D',C'SERVEUR INDISPONIBLE',X'0D'
MSG4F    EQU    *
MSG10    DC    Y(MSG10F-* -2)
          DC    X'0C',C'TAPEZ UN IDENTIFIANT PUIS ENVOI      '
          DC    X'0A',C'          M POUR UN MINITEL        '
          DC    X'0A',C'          P  POUR UN PC             '
MSG10F    EQU    *
          USSEND
          END

```

VTAM USS table

## 5.6. CICS Definitions

When a Minitel or VIRTEL Web Access terminal logs on via VIRTEL to CICS, the application relay LU represents the terminal as seen by CICS. The relay LU must therefore be referenced in the CICS CSD file, or alternatively configured by the AUTOINSTALL program of your site that will decide which TYPETERM to assign to which relay.

### 5.6.1. VIRTEL Web Access Terminals

The following example shows CSD definitions for VIRTEL Web Access terminals. The NETNAME parameter must match the "Relay" name specified in the definition of the VIRTEL terminals attached to the HTTP line. For more details, refer to the section entitled "Definition of an HTTP line" in the VIRTEL Configuration Reference documentation.

```

* VIRTEL 3270 TERMINALS FOR WEB2HOST
DEFINE  TERMINAL(T000) GROUP(VIRTEL) TYPETERM(DFHLU2E2)

```

```

      NETNAME(RHTVT000) PRINTER(I000)
      DESC(VIRTEL WEB TO HOST TERMINAL)
DEFINE  TERMINAL(T001) GROUP(VIRTEL) TYPETERM(DFHLU2E2)
      NETNAME(RHTVT001) PRINTER(I001)
      DESC(VIRTEL WEB TO HOST TERMINAL)
DEFINE  TERMINAL(T002) GROUP(VIRTEL) TYPETERM(DFHLU2E2)
      NETNAME(RHTVT002) PRINTER(I002)
      DESC(VIRTEL WEB TO HOST TERMINAL)
DEFINE  TERMINAL(T003) GROUP(VIRTEL) TYPETERM(DFHLU2E2)
      NETNAME(RHTVT003) PRINTER(I003)
      DESC(VIRTEL WEB TO HOST TERMINAL)

```

*CICS definitions for VIRTEL Web Access terminals*

### 5.6.2. Minitel Terminals

In order to access CICS applications, each Minitel must have an entry defined in the TCT or CSD. Specify the name of the relay LU associated with the terminal on the NETNAME parameter in the terminal definition.

```

MIN1 DFHTCT TYPE=TERMINAL,TRMTYPE=LUTYPE2,ACCMETH=VTAM, *
      FEATURE=(DCKYBD,UCTRAN),TCTUAL=255, *
      TRMIDNT=MIN1,TRMMODL=2,TRMSTAT=TRANSCIVE, *
      CHNASSY=YES,CONNECT=NO,GMMSG=NO,RUSIZE=1024, *
      TIOAL=(1024,4096),BUFFER=0,NETNAME=VIRTMINI

```

It is recommended that you do not use automatic initialisation of the 'good morning' message in the CICS TCT as it may conflict with VIRTEL's ability to call a specific CICS transaction.

```

OVERTYPE TO MODIFY                                CICS RELEASE=0330
CEDA Alter
  TYPeterm      : VIRTMINI
  Group         : VIRTEL
  DEScription   ==>
RESOURCE TYPE
  DEVice       ==> LUTYPE2
  TERModel     ==> 2
  SESSiontype  ==>
  LDclst       :
  SHippable    ==> No                No ! Yes
MAPPING PROPERTIES
  PAGesize     ==> 024 , 080        0-999
  ALTPage      ==> 000 , 000        0-999
  ALTSuffix    ==>
  FMhparm      ==> No                No ! Yes
  OBOperid     ==> No                No ! Yes
PAGING PROPERTIES
  AUTOPage     ==> No                No ! Yes
  DEFscreen    ==> 024 , 080        0-999
  ALTScreen    ==> ,                0-999
  APLKybd      ==> No                No ! Yes
  APLText      ==> No                No ! Yes
  AUDiblealarm ==> No                No ! Yes
  COLor        ==> Yes              No ! Yes
  COPy         : No                No ! Yes
  DUAlcasekybd ==> No                No ! Yes
  EXTendedds   ==> Yes              No ! Yes
  HILight      ==> Yes              No ! Yes
  Katakana     ==> No                No ! Yes
  LIghtpen     ==> No                No ! Yes
  Msrcontrol   ==> No                No ! Yes
  OBFormat     ==> No                No ! Yes
  PARTitions   ==> No                No ! Yes
  PRIntadapter ==> No                No ! Yes
  PROgsymbols  ==> No                No ! Yes
  VALidation   ==> No                No ! Yes

```

```

Formfeed      ==> No                No ! Yes
HOrizform     ==> No                No ! Yes
VErticalform  ==> No                No ! Yes
TEXTKybd     ==> No                No ! Yes
TEXTPrint    ==> No                No ! Yes
Query        ==> No                No ! Cold ! All
Outline      ==> No                No ! Yes
SOsi         ==> No                No ! Yes
BACktrans    ==> No                No ! Yes
CGcsgid      ==> 00000 , 00000     0-65535
SESSION PROPERTIES
AScii        ==> No                No ! 7 ! 8
SENdsize     ==> 00000             0-30720
RECEivesize  ==> 00256             0-30720
BRacket      : Yes                Yes ! No
LOGMode      ==>
DIAGNOSTIC DISPLAY
ERRLastline  ==> No                No ! Yes
ERRIntensify ==> No                No ! Yes
ERRColor     ==> NO
ERRHilght    ==> No                No ! Blink !Reverse !Und
OPERATIONAL PROPERTIES
AUTOCconnect ==> No                No ! Yes ! All
ATi          ==> Yes                No ! Yes
TTi          ==> Yes                Yes ! No
CReatesess   ==> Yes                No ! Yes
RELreq       ==> Yes                No ! Yes
DIScreq      ==> Yes                Yes ! No
Nepclass     ==> 000               0-255
Signoff      ==> Yes                Yes ! No ! Logoff
Xrfsignoff   ==> Noforce            Noforce ! Force
MESSAGE RECEIVING PROPERTIES
Routedmsgs   ==>                  All ! None ! Specific
LOGOnmsg     ==> No                No ! Yes
APPLICATION FEATURES
BUildchain   : Yes                No ! Yes
USerarealen  ==> 0                 0-255
Ioarealen    ==> 04096 , 04096     0-32767
UCtran       ==> Yes                No ! Yes ! Tranid
RECOVERY
RECOVOption  ==> Sysdefault         Sysdefault!Clearconv!Releasesess!Uncondrel!None
RECOVNotify  ==> None                None ! Message
OVERTYPE TO MODIFY CICS RELEASE=0330
CEDA Alter
TERminal     : I408
GRoup        : VIRTEL
Description  ==>
AUTINSTModel ==> No                No ! Yes ! Only
AUTINSTName  ==>
TERMINAL IDENTIFIERS
TYpeterm     ==> VIRTMINI
NEtname      ==> VIRMI408
CONSOle      ==> No                No ! 0-127 ! 129-250
CONSName     ==>
REMOTESystem ==>
REMOTEName   ==>
Modename     ==>
ASSOCIATED PRINTERS
PRINTER      ==>
PRINTERCopy  ==> No                No ! Yes
ALTPRINTER   ==>
ALTPRINTCopy ==> No                No ! Yes
PIPELINE PROPERTIES
POol         ==>
Tasklimit    ==> No                No ! 1-32767
OPERATOR DEFAULTS
OPERId       :
OPERPriority  : 000                0-255
OPERRsl      : 0                  0-24,

```

```

OPERSecurity : 1          1-64,.
PRESET SECURITY
  Userid      ==>
  NATlang     ==>
TERMINAL USAGES
  Transaction ==>
  Termpriority ==> 000      0-255
  Inservice   ==> Yes      Yes ! No
SESSION SECURITY
  Securityname ==>
  ATtachsec    ==> Local    Local ! Identify ! Verify ! Persistent ! Mixidpe
  BINDPassword ==>          PASSWORD NOT SPECIFIED
  BINDSecurity ==> No       No ! Yes

```

*CICS definitions for Minitel terminals*

## 6. VIRTCT

### 6.1. Introduction

All the general information necessary for VIRTEL to run is contained in a table known as the VIRTCT. After initialising the different parameters, this table must be assembled and link edited with the name VIRTCTxx, where xx are the two characters that identify the VIRTCT at start up time to the system. This xx value will be contained in the parameter of the PARM operand of the VIRTEL start procedure in MVS, or behind the EXEC card in the VSE environment.

The VIRTCT must be assembled before VIRTEL can be run. At the time of the assembly the VIRTEL macro library VIRT4XX.MACLIB must be on-line. Options RENT and REUS must not be specified when assembling the VIRTCT for an MVS environment. The resulting phase or load module must be placed in the library containing the other phases or load modules required by VIRTEL.

For MVS, a sample VIRTCT source member is provided in the VIRTCT01 member of the VIRTEL SAMPLIB, and the assembly and link-edit JCL is in member ASMTCT. For VSE, a sample VIRTCT with assembly and link-edit JCL is in the VIRTCT installation job.

### 6.2. Parameters Of The VIRTCT

Some parameters have a default value taken by VIRTEL and do not need to be coded in your table.

#### 6.2.1. ACCUEIL parameter

ACCUEIL=(YES/NO[,KEEP])	Default=YES
-------------------------	-------------

##### YES

Terminals not defined in VIRTEL may be connected in ACCUEIL mode. That means the terminals will have access to all functions, excepting dialogue with another application (relay). The maximum number of terminals accepted in ACCUEIL mode is a function of the parameter of the operand NBDYNAM.

##### NO

Terminals not defined in VIRTEL may not be connected.



**KEEP**

Allows the Multi-Session screen to be used as a dynamic USSTAB without the terminals being associated with the application relays (See the heading 'Using the dynamic USSTAB' in the 'VIRTEL Multi-Session' chapter only available in French)

**6.2.2. ADDR1 parameter**

ADDR1=' '      Default=' '

The address line 1 of the client as specified in the key at the time of installation. This parameter is unique to each client and functions in relation to the following parameters ADDR2, COMPANY, LICENSE, EXPIRE and CODE

**6.2.3. ADDR2 parameter**

ADDR2=' '      Default=' '

The address line 2 of the client as specified in the key at the time of installation. This parameter is unique to each client and functions in relation to the following parameters ADDR1, COMPANY, LICENSE, EXPIRE and CODE

**6.2.4. AIC parameter**

AIC=APPLID/TRANSACT      Default=APPLID

This parameter determines the value returned by the APPLICATION-IS-CONNECTED condition of the CREATE-VARIABLE-IF tag (see "Signon and password management" in the VIRTEL Web Access Guide). This in turn affects the window title of the VIRTEL Web Access screen. The following values are possible:

**APPLID**

The tag returns the VTAM applid of the host application.

**TRANSACT**

The tag returns the external name of the VIRTEL transaction used to access the host application.

**6.2.5. ANNUL parameter**

ANNUL=xx      Default=6D (Clear)

**xx**

The 3270 AID function key which will be transmitted to the application when the Minitel user presses the [ANNULATION] key. This parameter allows the user to define a general parameter by default which may be modified in the definition of the sub-server nodes.

ANNUL=00 allows the cursor to be placed at the start of the field with erasure of the field.

**6.2.6. APPLID parameter**

APPLID=nappl      Default=VIRTEL

**nappl**

The name of the primary VIRTEL ACB.

The APPLID parameter specifies the label or ACBNAME parameter of the VTAM APPL for the primary VIRTEL ACB. The value specified here can be overridden in the VIRTEL startup JCL (see [“Executing VIRTEL in an MVS environment”](#), page 22 or [“Executing VIRTEL in a VSE environment”](#), page 40 for details).

When no primary VTAM ACB is required (for example, in the VIRTCT for a VIRTEL Batch job), then this parameter may be coded as APPLID=\*NOAPPL\*

If SYSPLUS=YES is specified, a '+' character in the APPLID will be replaced by the value of the SYSCONE system symbol. SYSCONE is specified in the IEASYMxx member of SYS1.PARMLIB, and identifies the particular LPAR that VIRTEL is running on in a sysplex environment.

### 6.2.7. APPSTAT parameter

APPSTAT=YES/NO      Default=NO

#### YES

The status of the applications (active or non active) is tested at the time of access to the VIRTEL Web Access Application Selection Menu and the VIRTEL Multi-Session screen. For VIRTEL Web Access the status of each application is indicated by a color (see [“Application Selection Menu”](#) in the VIRTEL Web Access Guide). For VIRTEL Multi-Session the test is based on the value contained in the [“STATUS”](#) field of the application definition screen. The function key allowing access to the application will only appear if the application is active.

#### NO

The function key allowing access to the application is always present.

### 6.2.8. ARBO parameter

ARBO=YES/NO      Default=NO

#### YES

The program for managing the Minitel tree structure will function as a VIRTEL internal sub-application.

#### NO

The tree structure management software will not function.

### 6.2.9. BATCH1 parameter

BATCH1=(indd,indcb,outdd,outdcb)      Default=no batch connection

This parameter defines the batch processing characteristics for all lines which specify type BATCH1.

#### indd

The batch input DD name (for example, SYSIN).

#### indcb

The label of the DCB macro defining the batch input file. This DCB macro must appear later in the VIRTCT (see [“Additional parameters for batch files”](#), page 79).

#### outdd

The batch output DD name (for example, SYSPRINT).

#### outdcb

The label of the DCB macro defining the batch output file. This DCB macro must appear later in the VIRTCT (see [“Additional parameters for batch files”](#), page 79).

### 6.2.10. BATCH2 parameter

BATCH2=(indd,indcb,outdd,outdc)      Default=no 2nd batch connection

This parameter defines the batch processing characteristics for all lines which specify type BATCH2. The subparameters are the same as those of the BATCH1 parameter.

### 6.2.11. BFVSAM parameter

BFVSAM=n      Default=8192

**n**

Size of VSAM buffer ("CI size") used by VIRTEL for reading files such as GTVSAM. As a general rule, this value is calculated by VIRTEL and should not be modified. The size is normally 8192.

### 6.2.12. BUFDATA parameter

BUFDATA=n      Default=16

**n**

The number of VSAM buffers in the pool allocated for file access.

### 6.2.13. BUFSIZE parameter

BUFSIZE=n      Default=8192

**n**

The size of the largest VTAM message that may pass through VIRTEL. Generally this value should not be modified. The size is generally 8192.

### 6.2.14. CHARSET parameter

CHARSET=(charset1,charset2,...)      Default=none

The CHARSET parameter allows tables of non-standard character sets to be loaded into VIRTEL at startup time. DBCS tables, because of their size, are not loaded by default into VIRTEL and must be explicitly requested using this parameter. The standard and non-standard tables are used for EBCDIC - UTF-8 translation and can be specified by the SET-OUTPUT-ENCODING-UTF-8 tag and by the DEFUTF8 parameter of the VIRTCT. Refer to the description of the DEFUTF8 parameter 55 for the list of standard tables which are always loaded into VIRTEL.

#### charset

The following non-standard tables can be loaded:

- IBM933A: Korean host mixed
- IBM1364: Korean host mixed extended
- IBM1388: Chinese simplified SBCS et DBCS
- IBM1390: Japanese Katakana-Kanji
- IBM1399: Japanese Latin-Kanji

### 6.2.15. CODE parameter

CODE='xxxxxxx'      Default='      '

#### xxxxxxx

Is the code calculated for the client as it is specified in the installation key at the time of the installation. This code is unique for each client and functions in relation to the following parameters: ADDR1, ADDR2, COMPANY, LICENSE, and EXPIRE.

### 6.2.16. COMPANY parameter

COMPANY='      '      Default='      '

The name of the company as it is specified in the installation key at the time of the installation. This code is unique for each client and functions in relation to the following parameters: ADDR1, ADDR2, LICENSE, EXPIRE and CODE.

### 6.2.17. COMPR3 parameter

COMPR3=NO/AUTO/FIXED      Default=NO

#### NO

Level 3 compression for PC's will not be used.

#### AUTO

Level 3 compression for PC's will be used. VIRTEL will run in learning mode as well as processing screen types.

#### FIXED

Level 3 compression for PC's will be used. VIRTEL will only run processing screen types.

### 6.2.18. CORRECT parameter

CORRECT=xx      Default=7C (PF12)

#### xx

The 3270 AID function key which will be transmitted to the application when the Minitel user presses the [CORRECTION] key in a blank field.

**CORRECT=00 places the cursor at the start of the field without sending anything to the application.**

### 6.2.19. COUNTRY parameter

COUNTRY=xxx      Default=FR

#### xxx

Country name indicating which translation table is to be used for translation between EBCDIC and ASCII when UTF-8 is not specified.

Possible values are:

Value	Country	EBCDIC Code Page	ASCII Code Page
ALBANIA	Albania	CP 500	8859-1
AUSTRALIA	Australia	CP 037	8859-1
BE or BELGIUM	Belgium	CP 500	8859-1
BRAZIL	Brazil	CP 037	8859-1
CANADA	Canada	CP 500	8859-1
DENMARK	Denmark	CP 277	8859-1
DE or GERMANY	Germany	CP 273	8859-1
FI or FINLAND	Finland	CP 278	8859-1
FR or FRANCE	France	CP 297	8859-1
IS or ICELAND (IC)	Iceland	CP 871	8859-1
IRELAND	Ireland	CP 285	8859-1
IT or ITALY	Italy	CP 280	8859-1
L2 or ISO-LATIN-2	Eastern Europe	CP 870	8859-2
NETHERLAND	The Netherlands	CP 037	8859-1
NO or NORWAY	Norway	CP 277	8859-1
PORTUGAL	Portugal	CP 037	8859-1
P2 or PC-LATIN-2	Eastern Europe	CP 870	CP 852
ES or SPAIN (SP)	Spain	CP 284	8859-1
SWEDEN	Sweden	CP 278	8859-1
SWITZERLAND	Switzerland	CP 500	8859-1
GB (UK)	United Kingdom	CP 285	8859-1
US or USA	United States	CP 037	8859-1

Note: The values shown in parentheses in the table above are accepted for compatibility with previous versions of VIRTEL.

The COUNTRY parameter is not used when displaying web pages which contain a {{{SET-OUTPUT-ENCODING-UTF-8}}} tag. In this case VIRTEL uses an EBCDIC-to-UTF-8 translate table determined by the “DEFUTF8 parameter”, page 55 or specified in the tag itself.

### 6.2.20. CRYPT1 parameter

```
CRYPT1=(name1,[algs],[algp],[engine],[encoding],[chaining],[padding])
      Default=none
```

This parameter defines the characteristics of the encryption performed by VIRTEL for page templates which specify the cryptographic identifier name1.

#### name1

A name which serves to identify this set of encryption parameters. This name will be referenced in the PUBLIC-KEY and ENCRYPTION-PARAMETERS tags in the HTML page template which uses encrypted fields.

#### algs

The symmetric encryption algorithm to be used by VIRTEL for data encryption. The following values can be specified:

**NONE**

| (default value) No encryption

**DES**

| Data Encryption Standard (8 byte key)

**2TDEA**

| Triple Data Encryption Algorithm, keying option 2 (16 byte key)

**3TDEA**

| Triple Data Encryption Algorithm, keying option 3 (24 byte key)

**AES-128**

| Advanced Encryption Standard, key size 128 bits (16 byte key)

**AES-192**

| Advanced Encryption Standard, key size 192 bits (24 byte key)

**AES-256**

| Advanced Encryption Standard, key size 256 bits (32 byte key)

**Note:** in this version of VIRTEL, only NONE, DES, 2TDEA, and 3TDEA are supported

**algp**

The asymmetric encryption algorithm to be used by VIRTEL for encryption of session keys. The following values can be specified:

**NONE**

| (default value) No encryption

**RSA-512**

| RSA public key encryption (512 bit key)

**RSA-1024**

| RSA public key encryption (1024 bit key)

**RSA-2048**

| RSA public key encryption (2048 bit key)

**RSA-4096**

| RSA public key encryption (4096 bit key)

**Note:** in this version of VIRTEL, only NONE, RSA-512, and RSA-1024 are supported

**engine**

The name of the encryption engine to be used. The following values can be specified:

**ICSF**

| VIRTEL uses the Integrated Cryptographic Service Facility of z/OS

**NO-ENCRYPTION**

| (default value) VIRTEL uses an internal null-encryption engine. In this case, NONE must be specified or defaulted for the cryptographic algorithms.

**encoding**

The representation which VIRTEL will use for encrypted text. The following values can be specified:

**HEX**

| (default value) Encrypted data is represented in hexadecimal

**BASE64**

| Encrypted data is represented in base64 format

Note: in this version of VIRTEL, only HEX is supported

**chaining**

The chaining method to be used for symmetric encryption. The following values can be specified:

**CBC**

| (default value) Cipher block chaining will be used

**ECB**

| Electronic codebook will be used

Note: in this version of VIRTEL, only CBC is supported

**padding**

The padding method to be used for symmetric encryption. The following values can be specified:

**PKCS7**

| (default value) Public Key Cryptographic Standard #7 padding

**X9.23**

| ANSI X9.23 padding method

**ISO10126**

| Padding method using random padding bytes

Note: in this version of VIRTEL, only PKCS7 is supported

**6.2.21. CRYPT2 parameter**

```
CRYPT2=(name2,[algs],[algp],[engine],[encoding],[chaining],[padding])
      Default=none
```

This parameter defines the characteristics of the encryption performed by VIRTEL for page templates which specify the cryptographic identifier name2. The subparameters are the same as those of CRYPT1.

**6.2.22. DEFENTR parameter**

```
DEFENTR=(xxxxxxxx,yyyyyyyy)   Default=' '
```

**xxxxxxxx**

| The name of the entry point taken by default at connection time by a 3270 terminal. This parameter may for example be used for 3270 connections functioning in ACCUEIL mode.

**yyyyyyyy**

| The name of the default entry point for X25 asynchronous connections.

**6.2.23. DEFUTF8 parameter**

```
DEFUTF8=xxxxxxxx   Default=IBM1147
```

**XXXXXXX**

Name of the default character set for EBCDIC to UTF-8 translation. This character set is used when an HTML or XML page contains a SET-OUTPUT-ENCODING-UTF-8 tag without a character set name. Any one of the following values may be specified:

Character set	Description
IBM0037	US EBCDIC (without Euro sign)
IBM1047	Latin-1 Open Systems EBCDIC
IBM1140	ECECP USA, Canada, Netherlands, Portugal, Brazil, Australia, New Zealand
IBM1141	ECECP Austria, Germany
IBM1142	ECECP Denmark, Norway
IBM1143	ECECP Finland, Sweden
IBM1144	ECECP Italy
IBM1145	ECECP Spain, Latin America (Spanish)
IBM1146	ECECP United Kingdom
IBM1147	ECECP France UCS-2
IBM1148	ECECP International 1
IBM1149	ECECP Iceland
IBM1153	Latin-2 - EBCDIC multilingual with euro
IBM1154	Cyrillic multilingual with euro
IBM1155	Turkey Latin 5 with euro
IBM1156	Baltic multilingual with euro
IBM1157	Estonia EBCDIC with euro
IBM1158	Cyrillic Ukraine EBCDIC with euro
IBM1159	T-Chinese host extended SBCS with euro
IBM1160	IBM1160
IBM1164	EBCDIC Vietnamese with euro
IBM4971	Greek (including euro)
IBM5123	Japanese Latin host extended SBCS (includes euro)
IBM12712	Hebrew (max set including euro and new sheqel)
IBM16804	Arabic (all presentation shapes) with euro
IBM1137	Devanagari (Hindi) EBCDIC (based on Unicode character set)

The values listed above are the names of the standard tables which are always available in VIRTEL. Additional tables may be loaded at startup time by means of the “[CHARSET parameter](#)”, page 51.

#### 6.2.24. DIRECT parameter

DIRECT=xx      Default=1C (REPR0)

**xx**

Hex code of the character of the 3270 keyboard that will be used to switch directly from one session to another. If DIRECT=00 then this function will be disabled.



### 6.2.25. DONTSWA parameter (MVS only)

DONTSWA=YES/NO      Default=NO

#### YES

VIRTEL will attempt to set itself non-swappable. This option is only available if VIRTEL is run from an APF-authorized library.

#### NO

VIRTEL remains swappable

**Note:** When VIRTEL is executed via program VIR6000, it is always non-swappable

### 6.2.26. EXIT1 parameter

EXIT1=xx      Default=' '

#### xx

Is the name of the VIRExx module that will be called to process an incoming call packet. This exit will only function for lines running in GATE mode.

### 6.2.27. EXIT2 parameter

EXIT2=xx      Default=' '

#### xx

Is the name of the VIRExx module that will be called when a Minitel sub-server node connects. If the line used is set to GATE mode this exit will process call packet CUD.

### 6.2.28. EXIT3 parameter

EXIT3=xx      Default=' '

#### xx

Is the name of the VIRExx module that will be called at connection time to a VTAM application from a multi-session screen.

### 6.2.29. EXIT4 parameter

EXIT4=xx      Default=' '

#### xx

Is the name of the VIRExx module that will be used to filter messages when a VTAM application is accessed either from the multi-session screen or from a Minitel sub-server node.

### 6.2.30. EXIT5 parameter

EXIT5=xx      Default=' '

**XX**

**E**

## XX

e

## XX

e

(Y)

e

## YF

NO

e

## 58

**xxxxxxx**

Is the DD name of the file used to save screen images captured during an external server call. To enable the screen image capture facility, specify FCAPT=VIRCAPT and include a VIRCAPT DD/DLBL statement in the VIRTEL JCL procedure. If the FCAPT parameter is omitted, the screen image capture facility is disabled.

### 6.2.37. FCMP3 parameter

FCMP3=xxxxxxx	Default=VIRCMP3
---------------	-----------------

**xxxxxxx**

Indicates the DD name of the file containing the screen types used in level 3 compression. To enable the level 3 compression facility, specify FCMP3=VIRCMP3 and include a VIRCMP3 DD/DLBL statement in the VIRTEL JCL procedure. The COMPR3 parameter specifies the type of compression. If COMPR3=NO is specified then the FCMP3 parameter is ignored and the VIRCMP3 file is not required.

### 6.2.38. GATE parameter

GATE=GENERAL/NO	Default=GENERAL
-----------------	-----------------

#### GENERAL

Activates support for all types of terminal.

#### NO

Activates support for incoming Minitel calls only.

### 6.2.39. GMT parameter

GMT=(x,y) GMT=(x,SYSTZ) GMT=SYSTZ	Default=(0,2)
---	---------------

This parameter indicates the timezone adjustments which VIRTEL must take into account in order to generate the correct standard conformant timestamps in SMTP and HTTP headers. This parameter is also used to generate timestamps in local time for the VIRLOG and VIRSTAT files.

#### x

The first subparameter is the number of hours which must be added to the system TOD clock value to arrive at GMT. Negative values indicate that the TOD clock is ahead of GMT, positive values indicate that the TOD clock is behind GMT. For systems which run with TOD=GMT this subparameter is 0.

#### y

The second subparameter is the number of hours which must be added to GMT to arrive at the local time. Negative values indicate that local time is behind GMT (west), positive values indicate that local time is ahead of GMT (east).

For example, USA EASTERN DAYLIGHT SAVINGS TIME with the TOD clock set to GMT should be coded as GMT=(0,-4). If the TOD clock is set to CENTRAL EUROPEAN TIME (GMT+1) and the local time is EUROPEAN SUMMER TIME (GMT+2) then this parameter should be coded as GMT=(-1,+2). GMT=(-1,+1) indicates that both TOD clock and local time are CENTRAL EUROPEAN TIME.

To avoid the need to modify the GMT parameter when daylight savings time is in effect, you may specify GMT=SYSTZ or GMT=(x,SYSTZ)

**GMT=SYSTZ**

indicates that the TOD clock is set to GMT and that VIRTEL will obtain the timezone difference by inspecting the system local time offset. For z/OS the local time offset is specified in the CLOCKxx member of the system PARMLIB, which may be modified by the SET CLOCK command in the event of a transition between winter and summer time. For VSE the local time offset is specified by the SET ZONEDEF command in the \$IPL procedure.

**GMT=(x,SYSTZ)**

indicates that the TOD clock is set to GMT-x, and VIRTEL will use the system local time offset to calculate the timezone difference. In this case, x is the number of hours which must be added to the TOD clock value to arrive at GMT, and VIRTEL considers the local time to be GMT + w – x where w is the system local time offset. GMT=SYSTZ is equivalent to GMT=(0,SYSTZ).

**6.2.40. GRNAME parameter**

GRNAME=grname	Default=none
---------------	--------------

**grname**

The VTAM generic resource name for VIRTEL.

If GRNAME is specified, VIRTEL will identify itself to VTAM using the specified generic resource name. The VTAM generic resources function allows the assignment of a generic resource name to a group of application programs that all provide the same function. VTAM automatically distributes sessions among these application programs rather than assigning all sessions to a single resource. Note: Use of generic resources requires a coupling facility structure.

**6.2.41. GTLOAD parameter**

GTLOAD=nn	Default=0
-----------	-----------

**nn**

Indicates the number of GTM map load modules.

**6.2.42. GTPRFE1 parameter**

GTPRFE1=(x1,x2,...,xn)	Default=' '
------------------------	-------------

**xn**

Indicates the base screen codes used in the \$%F commands of GTM. Each code references one of the 'ym' prefixes defined in the GTPRFE2 parameter. The number of codes defined in GTPRFE1 may not exceed the number of prefixes defined in the GTPRFE2 parameter.

**6.2.43. GTPRFE2 parameter**

GTPRFE2=(y1,y2,...,ym)	Default=' '
------------------------	-------------

**ym**

Indicates base screen prefixes associated with the code 'xn' defined in the GTPRFE1 parameter. The number of prefixes defined in the GTPRFE2 parameter must equal the number of codes defined in GTPRFE1 + 1; the last position contains the prefix to be used if no code is specified in the \$%F command or if the specified code does not exist.

### 6.2.44. GTVSAM parameter

GTVSAM=(filename,keylen,rkp,acbcard) Default=' '

#### filename

| Is the name of the VSAM file containing the GTM maps when these are contained in a VMO file.

#### keylen

| length of the VSAM key

#### rkp

| position relative to zero of the key in the record

#### acbcard

| Name of the ACB macro referenced, if the VMO file is described by a UFILEn parameter in the VIRTCT.

### 6.2.45. GTVSKIP parameter

GTVSKIP=n Default='0'

#### n

| Is the displacement used to localise the data in the VSAM record being read.

### 6.2.46. GUIDE parameter

GUIDE=xx Default=F1 (PF1)

#### xx

| The 3270 AID function key which will be transmitted to the application when the Minitel user presses the [GUIDE] key. This parameter allows the definition of a general value by default that may be modified when defining the sub-server nodes.

GUIDE=00 allows the [GUIDE] key to display a pad offering further choices.

### 6.2.47. HTFORWD parameter

HTFORWD=(proxy1,proxy2,...) Default=none

#### (proxy1,...)

| Specifies the IP address(es) of one or more proxy servers which forward HTTP requests to VIRTEL on behalf of clients. For all requests received from these proxies, VIRTEL obtains the client's IP address from the iv-remote-address: or the X-Forwarded-For: HTTP header generated by the proxy. This function may also be activated on a per-line basis by specifying the proxy address in the "Calling DTE" field of a rule (see "Rules" in the VIRTEL Connectivity Reference manual).

Note: IP addresses must include leading zeroes. For example, HTFORWD=(192.168.001.020,010.001.001.020)

### 6.2.48. HTHEADR parameter

HTHEADR=(h1,h2,...) Default=none

**(h1,h2,...)**

Specifies the names of up to 5 additional HTTP headers whose value is to be made available to scenarios. The names must be specified in upper case in this parameter, although the headers in the HTTP request may be upper or lower case. Refer to the description of the COPY\$ SYSTEM-TO-VARIABLE instruction in the VIRTEL Web Access Guide for further details.

**6.2.49. HTMINI parameter**

HTMINI=(len,time)      Default=(40,100)

The HTMINI parameter allows control over messages sent by VIRTEL Web Access applications. Certain applications may send several 3270 messages which together make up a complete screen. VIRTEL attempts to combine such messages into a single transmission to the browser, in order to avoid the need for the user to press ENTER to retrieve each message sent by the application.

VIRTEL considers that a message is possibly incomplete if the following conditions are true:

- The flag “restore keyboard” flag is not set in the 3270 WCC
- The “start printer” flag is not set in the 3270 WCC
- The message length is less than or equal to len bytes
- ~~The message does not contain an “insert cursor” command~~ [R.Bowler : "Text hidden because the presence of an insert cursor command no longer inhibits the operation of the HTMINI parameter (see updt2717 in Virtel 4.23)"]

After the arrival of a possibly incomplete message, VIRTEL waits for time hundredths of a second. If no other message has arrived during this interval, the possibly incomplete message is sent to the browser anyway. Otherwise, the possibly incomplete message is combined with the following message before sending it to the browser.

**6.2.50. HTPARM parameter**

HTPARM=(n1,n2)      Default=(30000,4096000)

This parameter allows you to override various VIRTEL Web Access settings. If HTPARM is specified, then all sub-parameters must be coded. The sub-parameters are:

**n1**

HTTP segment size. Do not change from the default value of 30000 unless advised by VIRTEL support.

**n2**

Maximum file size (in bytes) allowed for an IND\$FILE transfer. The default value 4096000 permits a maximum transfer size of approximately 4MB. For upload, if the size is exceeded the user will see HTTP response code “413 Request Entity Too Large”. For download, if the size is exceeded the user will see error message “TRANS14 Error reading file from host: file transfer canceled”.

**6.2.51. HTSET1 to HTSET4 parameters**

HTSETx=(option,option,...)      Default=none

These parameters allow various HTML processing options to be set as defaults. Each parameter has the form HTSETx = (option, option, ...) where option can take the values listed below:

**HTSET1**

MAXLENGTH, ID, BLANK-BINARY-ZEROES, HTML-ESCAPES, JAVASCRIPT-ESCAPES, XML-ESCAPES, AUTO-INCREMENT-VARIABLES

**HTSET2**

| NO-ADD-TO-CHECKBOX, NO-ADD-TO-LISTBOX, DO-NOT-IGNORE-BINARY-ZEROES

**HTSET3**

| Reserved for future use

**HTSET4**

| Reserved for future use

These processing options can be enabled or disabled within individual page templates via the SET-LOCAL-OPTIONS and UNSET-LOCAL-OPTIONS tags. Refer to the description of these tags in the VIRTEL Web Access Guide for the further details and for the meaning of each option.

**6.2.52. HTVSAM parameter**

HTVSAM=xxxxxxx      Default= (none)

**xxxxxxx**

Indicates the DD name in the VIRTEL JCL procedure of the VSAM file used to store the names of the e-mail correspondents for VIRTEL Web Access applications. Installations using the VIRTEL Web Access feature must specify HTVSAM=VIRHTML and include a VIRHTML DD/DLBL statement in the VIRTEL JCL procedure. If no HTTP or SMTP lines are defined in the VIRTEL configuration, then the HTVSAM parameter may be omitted, and the VIRHTML file is not required.

**6.2.53. IBERTEX parameter**

IBERTEX=YES/NO      Default=NO

**YES**

| Supports the CEPT1 (Spanish Minitel) standard.

**NO**

| Does not support the CEPT1 standard.

**6.2.54. IGNLU parameter**

IGNLU=(LuMch1,LuMch2,...)      Default=' '

**LuMchx**

The IGNLU parameter contains a list of line names which are not to be activated at VIRTEL startup time. When a line is referenced in this list, it can in no case be activated but is still manageable from the line management sub-application.

**6.2.55. LANG parameter**

LANG=' '/'E'      Default=' '

Specifies the language in which the VIRTEL administration panels are displayed. The following values are possible:

' '

| French language.

**'E'**

| English language.

**Note :** The apostrophes are required.

#### 6.2.56. LICENCE parameter

---

LICENCE=' '      Default=' '

Is the number of the licence attributed to the client as it is specified in the installation key at the time of the installation. This code is unique for each client and functions in relation to the following parameters: ADDR1, ADDR2, COMPANY, EXPIRE and CODE.

#### 6.2.57. LOCK parameter

---

LOCK=n      Default=32767

**n**

| Inactivity delay in minutes, after which a VIRTEL will lock a terminal and request the user to resubmit his password.

#### 6.2.58. MARK parameter

---

MARK=xx      Default=1E(EndField)

**xx**

| Code of the key enabling selection of fields in a Multi-Session copy / paste operation. The default key is 'end of field' : Shift PA2.

#### 6.2.59. MAXAPPL parameter

---

MAXAPPL=n      Default=64

**n**

| The maximum number of applications or transactions that may appear in the VIRTEL Multi-Session screen. The maximum value allowed is 64.

#### 6.2.60. MEMORY parameter

---

MEMORY=(BELOW/ABOVE[,DEBUG])      Default=BELOW  
MEMORY=NATIVE/TEST

Indicates the type of memory management used by VIRTEL

**BELOW**

| Memory managed by VIRTEL, with memory obtained below the 16 megabyte line.

**ABOVE**

| Memory is managed by VIRTEL, with memory obtained above the 16 megabyte line.



**NATIVE**

| Memory managed by MVS or VSE

**TEST**

| NATIVE plus ability to track memory usage.

MEMORY=ABOVE is recommended under MVS. MEMORY=(ABOVE,DEBUG) consumes more resources and is intended for debugging of memory corruption errors. NATIVE may produce a smaller real storage footprint for some HTML applications with very large numbers of terminals defined. TEST allows monitoring of memory usage by module via sub-application F4. TEST also produces a report of allocated memory via the output of the SNAP command.

**6.2.61. MINITEL parameter**

MINITEL=YES/NO      Default=YES

**YES**

| The Minitel environment for outgoing call handling will be established.

**NO**

| No Minitels and no outgoing calls.

**6.2.62. MQ1 parameter**

MQ1=(mqmname, 'prefix', [pgmname])      Default=no MQ connection

This parameter defines the characteristics of the connection to the message-queue manager (MQSeries) used by all lines which specify type MQ1.

**mqmname**

| The name of the message-queue manager (for example, CSQ1).

**prefix**

| A prefix which VIRTEL will add to all queue names. The prefix must be specified in quotes.

**pgmname**

| The name of the VIRTEL MQ interface program used for this connection. The following values can be specified:

**VIR0Q09**

| Interface program for MQSeries. This is the default.

**6.2.63. MQ2 parameter**

MQ2=(mqmname, 'prefix', [pgmname])      Default=no 2nd MQ connection

This parameter defines the characteristics of the connection to the message-queue manager (MQSeries) used by all lines which specify type MQ2. The subparameters are the same as those of the MQ1 parameter.

**6.2.64. MULTI parameter**

MULTI=YES/NO      Default=YES

**YES**

| Support for VIRTEL Multi-Session environment.

**NO**

| No Multi-Session.

**6.2.65. NBCVC parameter**

---

NBCVC=n	Default=8
---------	-----------

**n**

| The number of logical channels that are available for processing by VIRTEL.

**6.2.66. NBDYNAM parameter**

---

NBDYNAM=(t1,t2)	Default=(20,0)
-----------------	----------------

**t1**

| The number of 3270 terminals that may connect via a “dynamic terminal definition entry” (welcome mode).

**t2**

| The number of Minitel terminals that may connect via a “dynamic terminal definition”.

**6.2.67. NBTERM parameter**

---

NBTERM=nbterm	Default=500
---------------	-------------

**nbterm**

| Number of terminals envisaged running in VIRTEL. This parameter allows the user to estimate the maximum number events that may be waiting for service at any one time.

**6.2.68. NUMTASK parameter**

---

NUMTASK=nn	Default=4
------------	-----------

**nn**

| The number of primary tasks waiting events on the primary VIRTEL ACB.

**6.2.69. OTMAPRM parameter**

---

OTMAPRM=(exitname,userid,group,password,applname)	Default=( *SAMPLE* )
---	----------------------

This parameter defines the data which is passed to OTMA/IMSCoconnect in the header of a RESUME TPIPE request. All of the subparameters are optional. Missing subparameters are indicated by a comma.

**exitname**

| The identifier of the OTMA exit routine. Typical values are \*SAMPLE\* or \*SAMPL1\*. If omitted, the default value is \*SAMPLE\*.

**userid, group, password, applname**

Security parameters which VIRTEL will place in the userid, group, password, and application name fields in the RESUME TPIPE header.

**6.2.70. OSCORE parameter**

OSCORE=n                      Default=384

**n**

The number of kilobytes reserved for memory allocation by the operating system (e.g. for loading sub application modules). The default value of this parameter is calculated when this macro is assembled and is indicated by an MNOTE being issued. This value may optionally be reduced but a problem may then arise if all functions of the sub applications are used.

**6.2.71. PACKET parameter**

PACKET=n                      Default=128

**n**

The size of the packets used for transfer over the packet switched network.

**6.2.72. PASSTCK parameter**

PASSTCK=YES                      Default=none

This parameter activates PassTicket support in VIRTEL. The following values are possible:

**YES**

VIRTEL may generate PassTickets for VIRTEL transactions which specify 1 or 2 in the PassTicket field

If the PASSTCK parameter is omitted, VIRTEL will not generate PassTickets.

**6.2.73. PREZ900 parameter**

PREZ900=YES/NO                      Default=NO

Allows VIRTEL to run on a pre-zSeries processor. Possible values are:

**YES**

Specifies that the processor is a 9672, MP2000, MP3000, IS/390, or P/390. VIRTEL will not use instructions which are only available on z900 or later processors. Note: VIRTEL does not support 9672-G1, ES/9000, or any earlier processor.

**NO**

(default) Specifies that VIRTEL may use all instructions available on z900 or later processors.

**6.2.74. PRFSECU parameter**

PRFSECU='xxxxxxxx'                      Default=

**XXXXXXX**

Indicates the maximum 8 character prefix associated with the resources defined in the security management system if using RACF, TOP SECRET or ACF2.

### 6.2.75. PWPROT parameter

---

PWPROT=YES/NO	Default=NO
---------------	------------

**YES**

Supports protected field (DARK field) for 80 column Minitels with PAD=INTEG coded. This parameter must also be specified in NPSI.

**NO**

No support for the protected field (DARK field) for 80 column Minitels if PAD=INTEG.

### 6.2.76. RACAPPL parameter

---

RACAPPL=NONE/APPLID/GRNAME/'name'	Default=NONE
-----------------------------------	--------------

The RACAPPL parameter specifies the VIRTEL application name as it is known to RACF. When RACAPPL is present in the VIRTCT, VIRTEL will add an APPL= parameter to all RACROUTE VERIFY calls. RACF uses this application name for two purposes: (1) to validate that the user is authorized to access the VIRTEL application (if the RACF APPL class is active) and (2) to validate the user's passticket using the PTKTDATA class (if the user signs on to VIRTEL using a passticket instead of a password). The possible values are:

**NONE (or omitted)**

VIRTEL will not use the APPL= parameter on RACROUTE VERIFY calls. In this case RACF will use the default application name (MVSxxxx where xxxx is the 4-character SMF identifier of the z/OS system)

**APPLID**

VIRTEL will use the VTAM APPLID of the VIRTEL started task (specified in the APPLID parameter of the VIRTCT) as the value of the APPL= parameter for RACF.

**GRNAME**

VIRTEL will use the VTAM generic resource name of the VIRTEL started task (specified in the GRNAME parameter of the VIRTCT) as the value of the APPL= parameter for RACF. This setting may be useful in a sysplex environment. It allows all VIRTEL STCs in the sysplex to present the same application name to RACF.

**'name'**

VIRTEL will use the specified name as the value of the APPL= parameter for RACF. The name must be specified in single quotes.

### 6.2.77. RAPPL parameter

---

RAPPL=rappl	Default=USERA
-------------	---------------

**rappl**

Name of the security management resource class which contains the applications resources for the Multi-Session function and for external servers. The entities in this resource class are external servers and VTAM applications. If resource \$\$ALLSRV is used, then all the servers defined in VIRTEL are authorised.

### 6.2.78. REALM parameter

REALM=TRANSACTION/APPLID/GRNAME      Default=TRANSACTION

This parameter specifies the name presented by VIRTEL to the browser in the HTTP basic security signon dialog. The possible values are:

#### TRANSACTION

the external name of the VIRTEL transaction which is requesting security. This causes the browser to issue the signon prompt for each transaction the first time the transaction is requested in a browser session.

#### APPLID

the VTAM APPLID of the VIRTEL started task. With this setting VIRTEL presents the same realm name for all transactions, and thus the user sees only one signon prompt per browser session.

#### GRNAME

the VTAM generic resource name of the VIRTEL started task. This setting may be useful in a sysplex environment. It allows all VIRTEL STCs in the sysplex to present the same realm name to the browser.

### 6.2.79. REPET parameter

REPET=xx      Default=F2 (PF2)

#### xx

The 3270 AID function key which will be transmitted to the application when a Minitel user presses the [REPETITION] key. This parameter allows the definition of a general value by default which may be modified in the sub-server node definition. A value of 00 indicates that the [REPETITION] key will not be transmitted.

### 6.2.80. RESO parameter

RESO=YES/NO      Default=YES

#### YES

VIRTEL's VTAM network management sub-application will be activated.

#### NO

The network management sub-application will not be used.

### 6.2.81. RETOUR parameter

RETOUR=xx      Default=00

#### xx

The 3270 AID function key which will be transmitted to the application when the Minitel user presses the [RETURN] key. By default the [RETURN] key is not transmitted to the application but serves to set the cursor to the beginning of the preceding field. This parameter allows for the definition of a general value by default that may be modified in the definition of the sub-server nodes.

### 6.2.82. RNODE parameter

RNODE=rnode      Default=USERB

#### **rnode**

The name of the security management resource class which contains Minitel tree structure nodes, VIRTEL sub-application names, internal names of transactions associated with entry points, and directory names for file transfer.

### 6.2.83. SECUR parameter

SECUR=NO/VIRTEL/RACF/TOPS/ACF2/RACROUTE Default=NO

For the MVS environment, the following options can be specified:

#### **NO**

No security software is used to control access.

#### **VIRTEL**

VIRTEL's internal security management feature is used.

#### **(RACROUTE,RACF)**

IBM's security management product is used (via SAF).

#### **(RACROUTE,TOPS)**

The TOP SECRET security management product is used (via SAF).

#### **(RACROUTE,ACF2)**

The ACF2 security management system is used (via SAF).

For the VSE environment, the following options can be specified:

#### **NO**

No security software is used to control access.

#### **VIRTEL**

VIRTEL's internal security management feature is used.

#### **RACROUTE**

(for z/VSE Version 3 or later) VIRTEL uses the VSE Basic Security Manager (via SAF), or the External Security Manager if specified in the VSE IPL parameters.

The following options are retained for compatibility with previous releases:

#### **RACF**

RACF without SAF.

#### **TOPS**

TOP SECRET without SAF.

#### **ACF2**

ACF2 with ACFDIAG (Only for VM). For MVS, this is treated as (RACROUTE,ACF2).

#### **RACROUTE**

Multi product interface security (via SAF).

If MEMORY=ABOVE, RACF without SAF and TOPS without SAF are not supported.

### 6.2.84. SILENCE parameter

SILENCE=YES/NO      Default=NO

#### YES

Inhibits the sending of asynchronous terminal connection and disconnection messages to the log (VIR0026W, VIR0028W, VIR0051I, VIR0052I, VIR0505I, VIR0507I, VIR1551I, VIRHT51I, VIRNA51I, VIRPF28I, VIRPF51I, VIRPF52I, VIRPF99I, VIRQ912W, VIRQ922W, VIRT912W, VIRT922W). Also, messages VIR0002W and VIR0914E will be suppressed on a REQSESS request.

#### NO

Enables the sending of asynchronous terminal connection and disconnection messages to the log.

The VIRTEL command SILENCE can be used to dynamically modify this parameter.

### 6.2.85. SNAPW parameter

SNAPW=80/132      Default=132

Indicates the default presentation format for SNAP and other dumps (80 or 132 columns). This parameter can be dynamically modified by the VIRTEL SNAPW command.

### 6.2.86. SOMMR parameter

SOMMR=xx      Default=00

#### xx

The 3270 AID function key which will be transmitted to the application when the minitel user presses the [SUMMARY] key. By default, the [SUMMARY] key is not transmitted to the application but serves to return the user to the tree structure. This parameter allows for the definition of a default which may be modified in the sub-server node definition. Where the value specified is a '01', use of the [SUMMARY] key sets the cursor on the first field to be entered in the current screen.

### 6.2.87. STATDSN parameter

STATDSN=(dsn1,dsn2,...)      Default=none

#### dsn1,...

Dataset names of the files to be used for recording statistics if the parameter STATS=MULTI is specified. From 2 to 10 datasets can be specified. The datasets must be cataloged.

### 6.2.88. STATS parameter

STATS=YES/NO/(MULTI,CONTINUE/TERMINATE) /SMF/(SMF,nnn)      Default=YES

#### YES

Statistics recording is active. Statistics will be written to a single file defined in the VIRSTAT DD or DLBL statement in the VIRTEL started task JCL. VIRTEL must be stopped periodically to allow the statistics to be copied to a history file. The VIRSTAT file is overwritten each time VIRTEL is started.

**NO**

| Statistics will not be recorded.

**MULTI**

Statistics recording is active. Statistics are written to one of the datasets defined in the STATDSN parameter of the VIRTCT. VIRTEL rotates the datasets by switching automatically to the next dataset when the current dataset becomes full. A batch job can then be used to copy the statistics to an archive file without stopping VIRTEL. Message VIR0603I can be used by an automated operator to trigger the submission of the batch job. After copying the statistics, the batch job must empty the dataset by writing an EOF marker at the beginning, which allows VIRTEL to reuse the dataset (see member STATCOPY in the VIRTEL SAMPLIB).

The VIRTEL STAT command allows the console operator to display the status of the VIRSTATx datasets, or to force VIRTEL to switch to the next statistics dataset.

If all of the statistics datasets are full, there are two options:

**STATS=(MULTI,TERMINATE)**

| VIRTEL terminates, to avoid the possibility of losing any further statistics

**STATS=(MULTI,CONTINUE)(default)**

| VIRTEL continues, without recording any further statistics. To restart statistics recording, run a STATCOPY batch job to empty at least one VIRSTATx dataset, then issue the STAT,I command.

The STATS=MULTI option is only available in the MVS environment.

**SMF**

Statistics recording is active. Statistics are written into SMF.

The VIRTEL STAT command allows the console operator to display the status of the VIRSTATx datasets, or to force VIRTEL to switch to the next statistics dataset.

The SMF record format is the same as the current STATS record but prefixed by the standard SMF header.

**STATS=SMF**

| The default SMF record number is 223.

**STATS=(SMF,nnn)**

| The SMF record number used will be nnn. The specified number must be between 128 and 255.

The STATS=SMF/(SMF,nnn) option is only available in the MVS environment.

### 6.2.89. STRNO parameter

---

STRNO=n	Default=8
---------	-----------

**n**

| Number of concurrent accesses to VSAM files.

### 6.2.90. SUITE parameter

---

SUITE=xx	Default=00
----------	------------

**xx**

| The 3270 AID function key which will be transmitted to the application when the Minitel user presses the [SUITE] function key. By default the [SUITE] function key is not transmitted to the application but serves to set the cursor to



the following field. This parameter allows the definition of a general value by default that may be modified in the definition of the sub server node.

### 6.2.91. SWAP parameter

SWAP=Pnn                      Default=P24

#### Pnn

Identifies the 3270 function key that causes VIRTEL to return to the multi-session menu (for SNA terminals, the ATTN key also performs this function). This parameter may take the following parameter values P1 to P24, PA1, PA2, or CLR.

### 6.2.92. SYSPLUS parameter

SYSPLUS=YES/NO                      Default=NO

#### YES

VIRTEL will retrieve certain system symbols from z/OS. Whenever the '+' character appears in the APPLID parameter or in a terminal relay name, VIRTEL will replace the '+' by the value of the SYSCONE symbol.

#### NO

System symbols will not be retrieved, the '+' character will not be substituted in LU names, and the xxx-SYMBOL functionality of the NAME-OF tag and the COPY\$ SYSTEM-TO-VARIABLE instruction is not active (see VIRTEL Web Access Guide).

### 6.2.93. TCP1 parameter

TCP1=tcpname                      Default=no TCP/IP connection  
TCP1=( [tcpname] , , , [maxsock] , [pgmname] , [adsname] )

This parameter defines the characteristics of the connection to the TCP/IP stack used by all lines which specify type TCP1.

#### tcpname

The name of the TCP/IP stack. This name should match the TCPIPJOBNAME parameter in the TCPIP.TCPIP.DATA file of the TCP/IP stack, or the name of the TCP/IP started task itself if TCPIPJOBNAME is not specified. The value ANY indicates that a connection can be established with any TCP/IP stack. This parameter is ignored by the TCP/IP for VSE stack.

#### maxsock

In MVS, this is the maximum number of sockets for each type TCP1 line defined in VIRTEL. If this subparameter is not specified, TCP/IP determines the number (50 by default). The maximum value allowed by VIRTEL is 65535. However, for customers using older versions of MVS (z/OS V1R4 or earlier), the TCP/IP stack enforces an upper limit of 2000 on this subparameter. Also, the value of the MAXFILEPROC parameter in PARMLIB member BPXPRMxx must exceed the maxsock value.

In VSE, this is the total maximum number of sockets for all VIRTEL lines of type TCP1. The TCP/IP for VSE stack currently ignores the value specified here, and uses a fixed value of 8001 instead.

#### pgmname

The name of the VIRTEL TCP/IP interface program used for this connection. The following values can be specified:

#### VIR0T09

Interface program using ASYNC=EXIT mode. This is the default for MVS systems.

**VIR0T10**

| Interface program using ASYNC=ECB mode. This is the default for VSE systems.

**adsname**

| The name which VIRTEL uses to identify itself to TCP/IP. The value \* indicates that VIRTEL uses its VTAM APPLID as the address space identifier. The default value is blank, which means that TCP/IP will assign the name of the VIRTEL started task as the address space identifier. This parameter is ignored by the TCP/IP for VSE stack.

**6.2.94. TCP2 parameter**

TCP2=tcpname                      Default=no 2nd TCP/IP connection  
TCP2=([tcpname],,,[maxsock],[pgmname],[adsname])

This parameter defines the characteristics of the connection to the TCP/IP stack used by all lines which specify type TCP2. The subparameters are the same as those of TCP1.

**6.2.95. TIMEOUT parameter**

TIMEOUT=n                      Default=5

**n**

| Indicates in minutes the time-out after which a terminal connected to an external server will be force disconnected if no line activity is seen. A value of 0 means that the terminal will not be disconnected even if no activity is detected. The value specified here applies only when the "User time out" field in the external server definition is set to zero (see "Parameters of the external server" in the VIRTEL Connectivity Reference manual).

**6.2.96. TIMERQS parameter**

TIMERQS=(n1,n2,n3,n4)                      Default=(5,10,5,0)

This parameter indicates the timeout values (in seconds) used by VIRTEL when attempting to establish an outbound connection using the Application-to-Application API (FA29 structured field). If this parameter is specified, then all four sub-parameters must be coded. The sub-parameters are:

**n1**

| Timeout for VTAM connections.

**n2**

| Timeout for X25 connections.

**n3**

| Timeout for TCP/IP connections.

**n4**

| Reserved for future use.

**6.2.97. TITRE1 parameter**

TITRE1='cccc'                      Default='SYSPERTEC'

**cccc**

| The first line of the Multi-Session menu screen.

### 6.2.98. TITRE2 parameter

TITRE2='cccc'      Default='      '

#### cccc

| The second line of the Multi-Session menu screen.

### 6.2.99. TRACALL parameter

TRACALL=(p1,p2,...)      Default=none

#### p1,p2,...

Additional categories of trace data to be included in the VIRTEL internal trace. One or more of the following values may be coded in any order:

#### HTTP

| Additional trace data for HTTP server

#### VSAM

| Additional trace data for VSAM I/O requests

#### XM

| Additional trace data for Cross-Memory communication

### 6.2.100. TRACBIG parameter

TRACBIG=n      Default=40

#### n

The number of entries reserved for the VIRTEL internal trace. The value indicated corresponds to n times 256 entries.

### 6.2.101. TRACEB parameter

TRACEB=nn      Default=200

#### nn

| The number of 1K buffers reserved for buffer data associated with entries in the VIRTEL internal trace. From VIRTEL 4.20 onwards, trace data is allocated above the 16MB line if possible.

### 6.2.102. TRACEOJ parameter

TRACEOJ=STANDARD/YES/NO      Default=NO

#### STANDARD

| An automatic SNAP of the VIRTEL internal trace table will be produced at the start of VIRTEL termination.

#### YES

| An automatic SNAP of the VIRTEL internal trace table will be produced at the end of VIRTEL termination.

**NO**

| No SNAP at VIRTEL termination.

**6.2.103. TRACEON parameter**

---

TRACEON=YES/NO	Default=YES
----------------	-------------

**YES**

| The VIRTEL internal trace is active.

**NO**

| The VIRTEL internal trace is not active.

**6.2.104. TRACTIM parameter**

---

TRACTIM=CPU/LOCAL/TOD	Default=LOCAL
-----------------------	---------------

VIRTEL uses the TOD clock to timestamp each entry in its internal trace table. This parameter specifies whether or not the SNAP command should adjust the timestamps to match the local time used in the system message log. Possible values are:

**CPU**

| The last column in the SNAP trace, instead of the local time or TOD time in 1/10000 of a second, contains the total used CPU time for the current TCB in 1/10000 of a second.

CPU option only works on z/OS and on a machine having support for the ECTG (Extract CPU Time) instruction (i.e. Z9-109 and above).

**LOCAL**

| The SNAP command adjusts the timestamps in the internal trace table so that they display as local time. This is the recommended setting.

**TOD**

| Timestamps are not adjusted for local time.

**6.2.105. TRAN parameter**

---

TRAN=EVEN/ODD/NO	Default=Even
------------------	--------------

This parameter should be coded in the same way as for the X25MCH macro in NPSI.

**6.2.106. UFILE1 to UFILE20 parameters**

---

UFILEx=(ddname,acbcards,rkp,keylen,mode) Default=' '
--

These parameters define the VSAM files used by VIRTEL for HTML directories. Each parameter has the form UFILEx = (ddname, acbcards, rkp, keylen, mode) where:

- ddname is the DD name of the file as specified in the VIRTEL start procedure.
- acbcards is the name of the ACB macro defining the access characteristics of the transfer file. This ACB macro must appear later in the VIRTCT (see [“Additional parameters for VSAM files”](#), page 79).

- rkp represents the relative position of the key in hexadecimal. This value must match the value specified in the DEFINE CLUSTER.
- keylen represents the length of the key in hexadecimal. This value must match the value specified in the DEFINE CLUSTER.
- mode represents one of the following values (in hexadecimal):

**00**

| read-only mode

**01**

| read/write mode

**05**| read-only mode or read/write mode, depending on the value of the “[VSAMTYP parameter](#)”, page 77 in the VIRTCT).

The UFILEx parameters must be defined in sequence with no intervening gaps in the suffix number x.

Sharing files explanation can be found here “[How to share VSAM files between multiple instances of VIRTEL](#)”, page 80.

### 6.2.107. VIRSECU parameter

VIRSECU=YES/NO

Default=NO

#### YES

The VIRTEL internal security sub-application is available. To use VIRTEL security, specify VIRSECU=YES with SECUR=VIRTEL. The combination SECUR=NO, VIRSECU=YES allows online definition of Virel security without security being active.

#### NO

| VIRTEL internal security is not available.

### 6.2.108. VIRSV1 parameter

VIRSV1=(vsvname)

Default=none

This parameter defines the characteristics of the interface to the VIRSV service request manager for service programs called from a scenario via the VIRSV\$ instruction.

#### vsvname

| Name of the service request manager. Must be VIRSV.

### 6.2.109. VSAMTYP parameter

VSAMTYP=READONLY

Default=NORMAL

#### READONLY

This parameter, if specified in the VIRTCT, allows the VIRTEL started task to be run in read-only mode for VSAM files, to be used in production mode, especially in a sysplex environment. Except for the VIRSWAP work file, all other VSAM files used by VIRTEL can be opened in read-only mode.

**NORMAL**

By default VIRTEL opens files in read/write mode to allow the possibility of updating certain VSAM files, such as the VIRARBO file for example.

**Note:** VSAMTYP=READONLY takes effect only if the appropriate values have been specified in the MACRF parameter of the ACB (see “Additional parameters for VSAM files”, page 79) and in the MODE subparameter of the UFILEx parameter of the VIRTCT (see “UFILE1 to UFILE20”, page 76).

**6.2.110. VTKEYS parameter**

VTKEYS=xxxxxxx      Default=0

**xxxxxxx**

The name of a table added to the end of the VIRTCT allowing for redefinition of the function keys for VT100. Please refer to the member VTSAMPLE in SAMPLIB.

**6.2.111. VTOVER parameter**

VTOVER=xxxxxxx      Default=none

**xxxxxxx**

The name of a table added to the end of the VIRTCT allowing for dynamic override of certain parameters in the VIRTCT. Please refer to the section “Dynamic VIRTCT overrides”, page 83 for further details.

**6.2.112. XM1 parameter**

XM1=xmname      Default=no XM connection  
XM1=(xmname,,, [pgmname])

This parameter defines the characteristics of the connection to the cross-memory manager (VIRXM) used by all lines which specify type XM1.

**Note:** Cross-memory is supported only on MVS systems (OS/390 or z/OS). The VIRXM product from Syspertec must also be installed.

**xmname**

The name of the cross-memory manager started task (VIRXM).

**pgmname**

The name of the VIRTEL XM interface program used for this connection. The following values can be specified:

**VIR0X09**

Interface program for MVS systems. This is the default.

**6.2.113. XM2 parameter**

XM2=xmname      Default=no 2nd XM connection  
XM2=( [xmname],,, [pgmname])

This parameter defines the characteristics of the connection to the cross-memory manager (VIRXM) used by all lines which specify type XM2. The subparameters are the same as those of the XM1 parameter.

### 6.2.114. ZAPH parameter

ZAPH=xxxxxxx      Default=none

#### xxxxxxx

The name of a table added to the end of the VIRTCT allowing for one or more patches to be applied at startup. Please refer to the section “[Applying patches via the VIRTCT](#)”, page 84 for further details.

## 6.3. Additional Parameters For VSAM Files

VIRTEL uses VSAM files for storing HTML pages and for VIRTEL/PC file transfer. These VSAM files must be defined in the VIRTCT by means of a parameter UFILEx and an ACBHx macro for each file. The ACB macros must be coded in the VIRTCT before the END card. The formats of these macros are as follows:

ACBHx      ACB      AM=VSAM, DDNAME=ddddddd, MACRF=(P1, P2, ..., Pn), STRNO=3

- The label ACBHx must match the acbname subparameter as specified in the UFILEx parameter of the VIRTCT.
- ddddddd represents the ddname of the file as it is known in the VIRTEL start up procedure.
- pn represents the authorisations granted to the transfer file. The permitted values are: SEQ, DIR, OUT, LSR (for read/write mode) or SEQ, DIR, IN, LSR (for read-only mode).
- The value OUT should be omitted from MACRF if you want the mode (read-only or read/write) to be determined by the value of the VSAMTYP parameter in the VIRTCT (see “[VSAMTYP parameter](#)”, page 77).

## 6.4. Additional Parameters For Batch Files

VIRTEL uses sequential files for batch input and output when the BATCHx parameter is defined in the VIRTCT, and a batch line is present in the configuration. These sequential files must be defined in the VIRTCT by means of a pair of DCB/DCBE macros for each file. The DCB/DCBE macros must be coded in the VIRTCT before the END card. The formats of these macros are as follows:

label	DCB	DDNAME=ddname, DCBE=labelx, DSORG=PS,	*
		LRECL=lrecl, RECFM=recfm, MACRF=(macrf)	
labelx	DCBE	EODAD=0, RMODE31=BUFF	

where:

#### label

corresponds to the DCB label specified in the BATCHx parameter

#### ddname

corresponds to the DD name specified in the BATCHx parameter.

For input files:

- lrecl is 80
- recfm is FB
- macrf is GL.

For output files:

- lrecl is 133
- recfm is FBA
- macrf is PM.

EODAD=0 should be specified for input files.

The example below shows how to code DCB/DCBE macros when the BATCH1 parameter is specified as BATCH1=(SYSIN1,DCBI1,SYSOUT1,DCBO1)

```

DCBI1    DCB    DDNAME=SYSIN1,      SYSIN DD          *
              DCBE=DCBI1X,          *
              LRECL=80,              *
              DSORG=PS,              *
              RECFM=FB,              *
              MACRF=( GL )           *
DCBI1X    DCBE  EODAD=0,RMODE31=BUFF
DCB01     DCB    DDNAME=SYSOUT1,     SYSPRINT DD       *
              DCBE=DCB01X,          *
              LRECL=133,            *
              DSORG=PS,            *
              RECFM=FBA,            *
              MACRF=( PM)           *
DCB01X    DCBE  RMODE31=BUFF

```

## 6.5. How To Share VSAM Files Between Multiple Instances Of VIRTEL

Some VSAM files are shareable between multiple instances of Virtel with the condition that a file can be opened in "write" mode by only one instance. File sharing can be implemented by modifying the corresponding UFILEx entry in the TCT and/or by using the VSAMTYP definition. Some files are not shareable, for example the statistics and swap files. These must be opened in read/write mode for each instance of Virtel.

For more detailed informations on this subject, see "[UFILE1 to UFILE20](#)", page 76 and "[VSAMTYP parameters of the VIRTCT, and also](#)", page 77 and "[Additional parameters for VSAM files](#)", page 79.

## 6.6. Example Of The VIRTCT

An example of the VIRTCT is supplied in member VIRTCT01 in the VIRTEL SAMPLIB for MVS, and in the installation job VIRTCT for VSE:

```

PRINT GEN
VIRTERM TYPE=INITIAL,APPLID=VIRTEL,          *
      COMPR3=AUTO,                            *
      LANG=' ',                               *
      COUNTRY=FR,                             *
      DEFUTF8=IBM1147,                        *
      CHARSET=,                               *
      GMT=SYSTZ,                              *
      CRYPT1=(CRYPT3270,NONE,NONE,NO-ENCRYPTION,HEX), *
      TCP1=(TCPIP,,250),                      *
      HTVSAM=VIRHTML,                        *
      BUFSIZE=20000,                         *
      ACCUEIL=YES,                          *
      DEFENTR=(PC,MINITEL),                  *
      FCAPT=VIRCAPT,                         *
      RETOUR=00,                             *
      SUITE=00,                              *

```



```

SOMMR=00,
CORRECT=00,
EXIT1=,
EXIT2=,
EXIT3=,
EXIT5=,
EXIT4=,
EXIT6=,
EXIT7=,
STATS=YES,          OU (MULTI,CONTINUE/TERMINATE)
STATDSN=(VIRTEL.STATA,VIRTEL.STATB),  SI STATS=MULTI
FCMP3=VIRCMP3,
APPSTAT=YES,
DONTSWA=YES,
NBDYNAM=250,
TRACEB=200,TRACEON=YES,TRACBIG=40,
MULTI=YES,RESO=YES,ARBO=YES,MINITEL=YES,
VIRSECU=YES,SECUR=NO,    VIRTEL,  (RACROUTE,RACF)
RAPPL=VIRTSERV,RNODE=VIRTNODE,
LOCK=20000,
TIMEOUT=5,
FASTC=NO,
UFILE1=(SAMPTRSF,ACBH1,0,10,01),
UFILE2=(HTMLTRSF,ACBH2,0,10,01),
UFILE3=(PLUGTRSF,ACBH3,0,10,01),
GATE=GENERAL,
NBCVC=32,
VTKEYS=VTTABLE,      VT100 : KEY REDEFINITION
MEMORY=ABOVE,
COMPANY='VOTRE COMPAGNIE      ',  VOIR LA CLE
ADDR1='VOTRE ADRESSE 1        ',  COMMUNIQUEE
ADDR2='VOTRE ADRESSE 2        ',  PAR SYSPERTEC
LICENCE='TYPE DE LICENCE ',
EXPIRE=(2999,12,31),
CODE='XXXXXXXX',
TITRE1='S Y S P E R T E C C O M M U N I C A T I O N ',
TITRE2='===== 4.2 ====='
* =====
VTTABLE  KTRANH ,          SAMPLE VT KEYS TRANSLATION
KTRAN KEY=D6D7,IS=F1,RETCODE=1    PF1
KTRAN KEY=D6D8,IS=F2,RETCODE=1    PF2
KTRAN KEY=D6D9,IS=F3,RETCODE=1    PF3
KTRAN KEY=D6E2,IS=F4,RETCODE=1    PF4
KTRAN KEY=D6E3,IS=F5,RETCODE=1    PF5
KTRAN KEY=D6E4,IS=F6,RETCODE=1    PF6
KTRAN KEY=D6E5,IS=F7,RETCODE=1    PF7
KTRAN KEY=D6E6,IS=F8,RETCODE=1    PF8
KTRAN KEY=D6E7,IS=F9,RETCODE=1    PF9
KTRAN KEY=D6E8,IS=7A,RETCODE=1    PF10
KTRAN KEY=D6E9,IS=7B,RETCODE=1    PF11
KTRAN KEY=D64A,IS=7C,RETCODE=1    PF12
KTRAN KEY=D6A7,IS=6D,RETCODE=1    CLEAR
KTRAN KEY=4AC8,IS=6D,RETCODE=1    CLEAR
*
KTRAN KEY=4AD2,IS=00,RETCODE=2    ERASEOF
KTRAN KEY=4AC1,IS=00,RETCODE=3    CURU
KTRAN KEY=4AC2,IS=00,RETCODE=4    CURD
KTRAN KEY=4AC3,IS=00,RETCODE=5    CURR
KTRAN KEY=4AC4,IS=00,RETCODE=6    CURL
* =====
ACBH1    ACB    AM=VSAM,DDNAME=SAMPTRSF,MACRF=(SEQ,DIR,OUT,LSR),
STRNO=3
ACBH2    ACB    AM=VSAM,DDNAME=HTMLTRSF,MACRF=(SEQ,DIR,OUT,LSR),
STRNO=3
ACBH3    ACB    AM=VSAM,DDNAME=PLUGTRSF,MACRF=(SEQ,DIR,OUT,LSR),
STRNO=3
END

```

Example VIRTCT

## 6.7. Assembling The VIRTCT

The VIRTCT must be assembled before starting VIRTEL for the first time. The VIRTEL macro library must be available to the assembler. In the MVS environment, the VIRTCT must be link-edited with the NORENT and NOREUS options. The RENT and REUS options must NOT be specified in the MVS environment. In the VSE environment, PRD1.MACLIB must be specified. The resulting phase or load module must be placed in a STEPLIB or SEARCH PHASE library available to the VIRTEL started task.

### 6.7.1. MVS example

A sample job for assembling the VIRTCT is supplied in member ASMTCT of the VIRTEL SAMPLIB:

```
//VIRTASM JOB 1,ASMTCT,MSGCLASS=X,CLASS=A,NOTIFY=&SYSUID
//*-----*
//* GENERATION VIRTCT ET EXIT DE VIRTEL *
//*-----*
//ASMTCT PROC OUTC=*,WORK=SYSDA,
//          QUAL=yourqual.VIRT453,
//          MEMBER=VIRTCT01
//*-----*
//* ASSEMBLAGE *
//*-----*
//ASM EXEC PGM=ASMA90,REGION=2048K,
//          PARM='NOOBJECT,DECK,XREF(SHORT),NORENT,LIST'
//SYSPRINT DD SYSOUT=&OUTC
//SYSLIB DD DSN=&QUAL..MACLIB,DISP=SHR
//          DD DSN=SYS1.MACLIB,DISP=SHR
//          DD DSN=SYS1.MODGEN,DISP=SHR
//SYSUT1 DD UNIT=&WORK,SPACE=(1700,(400,400))
//SYSUT2 DD UNIT=&WORK,SPACE=(1700,(400,400))
//SYSUT3 DD UNIT=&WORK,SPACE=(1700,(400,400))
//SYSPUNCH DD DSN=&LOADSET,UNIT=&WORK,DISP=(,PASS),
//            SPACE=(400,(100,100))
//SYSIN DD DSN=&QUAL..CNTL(&MEMBER),DISP=SHR
//*-----*
//* LINKEDIT *
//*-----*
//LKED EXEC PGM=HEWL,REGION=2048K,COND=(7,LT,ASM),
//          PARM='LIST,LET,XREF,NORENT'
//SYSPRINT DD SYSOUT=&OUTC
//SYSLIB DD DSN=&QUAL..LOADLIB,DISP=SHR
//SYSUT1 DD UNIT=&WORK,DCB=BLKSIZE=1024,SPACE=(1024,(200,20))
//SYSLIN DD DSN=&LOADSET,DISP=(OLD,DELETE)
//SYSLMOD DD DSN=&QUAL..LOADLIB(&MEMBER),DISP=SHR
//          PEND
//VIRTASM EXEC ASMTCT
```

*VIRTCT assembly in MVS*

### 6.7.2. VSE example

A sample job for assembling the VIRTCT is supplied on the installation tape:

```
* $$ JOB JNM=VIRTCT,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=Q
// JOB VIRTCT
// DLBL VIRT453,'VSE.VIRT453.LIBRARY',,VSAM,CAT=VSESPUC
// LIBDEF PHASE,CATALOG=VIRT453.SUBLIB
// LIBDEF SOURCE,SEARCH=(VIRT453.SUBLIB,PRD1.MACLIB)
// OPTION CATAL,NODECK,ALIGN
// PHASE VIRTCT01,*
// EXEC ASSEMBLY,SIZE=512K
```

```
* $$ SLI ICCF=(VIRTCT01),LIB=0037
/*
// EXEC LNKEDT,SIZE=512K
/*
/ &
* $$ E0J
```

*VIRTCT assembly in VSE*

## 6.8. Dynamic VIRTCT Overrides

Data may be passed to the VIRTEL procedure via the start command which allows the value of certain parameters in the VIRTCT (APPLID, MQ1, MQ2) to be modified. For example :

The VIRTEL started task procedure should contain the TCT, APPLID, and VTOVER parameters which are substituted into the PARM as shown below:

```
//VIRTEL EXEC PGM=VIR0000,
//          TIME=1440,REGION=8192K,
//          PARM=(&TCT,&APPLID,&VTOVER)
```

When starting VIRTEL, you may specify values for the TCT, APPLID, and VTOVER on the start command as shown in the example below:

```
S VIRTEL,TCT=01,APPLID=AA4AVIRX,VTOVER='12345'
```

After loading VIRTCT01, VIRTEL will then:

- replace the APPLID variable by the value AA4AVIRX
- use the value of the VTOVER variable to replace the specified substitution characters % in certain VIRTCT parameters by characters extracted from the VTOVER value.

This functionality requires that the VIRTCT should contain:

- a new parameter VTOVER=VTDYNAM
- a new table VTDYNAM consisting of macros VTOVERH and VTOVER
- the presence of one or more % characters in certain VIRTCT parameters which will be substituted by the corresponding characters from the VTOVER parameter specified on the start command.

For example, if the fourth character of the MQ1 and MQ2 Queue Manager name and the second three characters of the MQ2 Queue Name are variable, the parameters may be defined in the VIRTCT as shown below:

MQ1=(CSQ%),	-> wild char in MQ1 parm	*
MQ2=(CSQ%,'A%%'),	-> wild char in MQ2 parm	*
VTOVER=VTDYNAM,	-> new VIRTCT parm	*

At the end of the VIRTCT, define the VTDYNAM table with the rules for substituting data from the VTOVER parameter. For each parameter, specify the substitution character(s) to look for (TARGET), and the position relative to zero of the characters to be extracted from the VTOVER parameter (FROM), as shown in the example below:

VTDYNAM	VTOVERH	-> new table after the VIRTCT	
MQ1	VTOVER PARM=MQ1,	modify MQ1(1)	*
	TARGET='%',	find % char	*
	FROM=0,	replace % with VTOVER(0)	*
	ERRORC=12	Virtel RC if replace failed	
MQ21	VTOVER PARM=(MQ2,1),TARGET='%',FROM=1		
MQ22	VTOVER PARM=(MQ2,2),TARGET='%%',FROM=2		

With these definitions and VTOVER='12345', the MQ1 and MQ2 parameters of the VIRTCT will have the values shown below:

```
MQ1=CSQ1,
MQ2=(CSQ2, 'A345'),
```

If an error occurs during substitution, VIRTEL will issue message VIR0025E indicating the error code specified in the ERRORC parameter of the VTOVER macro.

## 6.9. Applying Patches Via The VIRTCT

The “[ZAPH parameter](#)”, page 0 of the VIRTCT allows one or more patches to be applied to the VIRTEL kernel after the resident modules have been loaded into memory at startup. This parameter is intended to be used only under the advice of Syspertec technical support personnel.

For example, if the VIRTCT contains the parameter:

```
ZAPH=MYPTFS, *
```

then it refers to the table MYPTFS coded after the VIRTERM macro, for example:

```
MYPTFS  ZAPH
USER4611 ZAPD VIR00TAB,+246D,EC,1B,' USERMOD TO TRANSLATE TABLE '
PTF4618  ZAPD VIR0011D,+092A,C98C,C984,' TEMP FIX FOR SEND$ '
```

The format of each ZAPD instruction is as follows:

```
label      ZAPD progname,+offset,verify,replace,'desc'
```

### label

| PTF identifier for message VIR0066I

### progname

| program name

### offset

| offset into program

### verify

| verify value (hexadecimal digits)

### replace

| replacement value (hexadecimal digits)

### desc

| (optional) description for message VIR0066I

# 7. VIRCONF

## 7.1. Introduction

The VIRCONF program allows a batch job to manage the VIRARBO file, which is the main configuration file for VIRTEL.

VIRCONF allows you to:

- Upload a new VIRARBO file using SYSIN cards
- Add, replace, or suppress one or more definitions within an existing VIRARBO file
- Create new définitions as SYSIN cards using an existing VIRARBO file
- Scan a SYSIN cards file for checking the right syntax

## 7.2. JCL

### 7.2.1. Creating and uploading a new VIRARBO file

Below are some JCL examples to define and upload a new VIRARBO file:

```

* $$ JOB JNM=VIRCONF,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=V,DEST=(,SPTUSER)
// JOB VIRCONF DEFINE AND LOAD VIRARBO
// DLBL IJSYSUC,'VSESP.USER.CATALOG',,VSAM
// EXEC IDCAMS,SIZE=AUTO
  DEFINE CLUSTER(NAME(VIRTEL.TESTARBO.KSDS) -
    RECORDS(500 100) SHAREOPTIONS(2 3) -
    RECSZ(600 4089) KEYS(9 0) -
    VOLUMES(DOSRES) TO(99366)) -
    DATA(NAME(VIRTEL.TESTARBO.KSDS.DATA)) -
    INDEX(NAME(VIRTEL.TESTARBO.KSDS.INDEX)) -
    CATALOG(VSESP.USER.CATALOG)
  IF LASTCC NE 0 THEN CANCEL JOB
/*
// LIBDEF *,SEARCH=(VIRT453.SUBLIB)
// DLBL VIRARBO,'VIRTEL.TESTARBO.KSDS',,VSAM,CAT=VSESPUC
// EXEC VIRCONF,PARM='LOAD'
(insert sysin control statements here)
/*
/&
* $$ E0J

```

*VIRCONF JCL in VSE to define and upload a new VIRARBO file*

```

//VIRCONF JOB CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID
/* THIS JOB DEFINES AND LOADS A NEW ARBO FILE
//DEFARBO EXEC PGM=IDCAMS,REGION=2M
//SYSPRINT DD SYSOUT=*
  DEFINE CLUSTER(NAME(VIRTEL.TEST.ARBO) -
    KEYS(9 0) RECSZ(100 4089) FSPC(10 10) -
    VOL(SPT001) REC(250,50) SHR(2) SPEED) -
    DATA(NAME(VIRTEL.TEST.ARBO.DATA) CISZ(4096)) -
    INDEX(NAME(VIRTEL.TEST.ARBO.INDEX))
//RELOAD EXEC PGM=VIRCONF,COND=(0,NE,DEFARBO),PARM=LOAD
//STEPLIB DD DSN=yourqual.VIRT453.LOADLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//VIRARBO DD DSN=VIRTEL.TEST.ARBO,DISP=SHR
//SYSIN DD DSN=&SYSUID.VIRCONF.SYSIN,DISP=SHR

```

*VIRCONF JCL in MVS to define and upload a new VIRARBO file*

When VIRCONF is executed with PARM=LOAD, control cards are read from SYSIPT (VSE) or SYSIN (MVS) and are loaded into the VIRARBO file.

### 7.2.2. Updating a VIRARBO file

Below are some JCL examples to add, replace, or delete one or more definitions from an existing VIRARBO file:

```

* $$ JOB JNM=VIRCONF,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=V,DEST=(,SPTUSER)
// JOB VIRCONF UPDATE VIRARBO
// LIBDEF *,SEARCH=(VIRT453.SUBLIB)
// DLBL VIRARBO,'VIRTEL4.VIRARBO.KSDS',,VSAM,CAT=VSESPUC
// EXEC VIRCONF,PARM='LOAD'
  LINE      ID=A-X0T,
            NAME=X0T-IP30,
            PARTNER=192.168.0.80:1998,
            LOCADDR=192.168.229.30:1998,
            DESC='Connections via Cisco router',
            TERMINAL=X0TF,INOUT=3,TYPE=TCP1,PROTOCOL=X0T,
            WINSZ=3,PKTSZ=128,RETRY=10,TIMEOUT=10,ACTION=0
  RULE      ID=AX200CFT,LINE=A-X0T,STATUS=ACTIVE,
            DESC="X0T->AntiPCNE->CFT (CUD0=X'C0')",
            ENTRY=APCFT,CUD0=(BEGIN,C0)
  DELETE    TYPE=RULE,ID=AX100CFT
/*

```

```
/&
* $$ E0J
```

*VIRCONF JCL in VSE to update a VIRARBO file*

```
//VIRCONF JOB CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID
/* THIS JOB UPDATES AN ARBO FILE
//UPDARBO EXEC PGM=VIRCONF,PARM=LOAD
//STEPLIB DD DSN=yourqual.VIRT453.LOADLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//VIRARBO DD DSN=VIRTEL.TEST.ARBO,DISP=SHR
//SYSIN DD *
DELETE TYPE=USER,ID=SAMPUSER
USER ID=BLOGGS,NAME='JOE BLOGGS',DEPT=VIRTEL,PASSWORD=JOE,
        PROFILE=(APPLICS,MINITEL,PC,REPRT,SECURITE,
        SERVEUR,SERVEXT,WEBMASTR)
/*
```

*VIRCONF JCL in MVS to update a VIRARBO file*

Submitting VIRCONF with PARM=LOAD for an existing VIRARBO file allows definitions to be added, replaced, or deleted, while keeping existing definitions in the VIRARBO file.

Using PARM='LOAD,NOREPL' parameter allows only new definitions to be added, while keeping existing definitions. In this case, VIRCONF will ignore any statement with the same name as existing definitions, returning a zero return code, except if another error was encountered.

### 7.2.3. Unloading a VIRARBO file

Below are some JCL examples to obtain existing VIRARBO definitions in the form of control cards:

```
* $$ JOB JNM=VIRCONF,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=V,DEST=(,SPTUSER)
* $$ PUN DISP=D,CLASS=W,DEST=(,SPTUSER)
// JOB VIRCONF UNLOAD VIRARBO TO SYSPCH
// LIBDEF *,SEARCH=VIRT453.SUBLIB
// DLBL VIRARBO,'VIRTEL.TESTARBO.KSDS',,VSAM,CAT=VSESPUC
// EXEC VIRCONF,PARM='UNLOAD'
/&
* $$ E0J
```

*VIRCONF JCL in VSE to unload a VIRARBO file*

```
//VIRCONF JOB CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID
/* THIS JOB UNLOADS AN ARBO FILE TO SYSPUNCH
//UNLOAD EXEC PGM=VIRCONF,PARM=UNLOAD
//STEPLIB DD DSN=yourqual.VIRT453.LOADLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//VIRARBO DD DSN=VIRTEL.TEST.ARBO,DISP=SHR,AMP=('RMODE31=NONE')
//SYSPUNCH DD DSN=&SYSUID.VIRCONF.SYSIN,DISP=(,CATLG),UNIT=SYSDA,
//          SPACE=(TRK,(5,1)),DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080)
```

*VIRCONF JCL in MVS to unload a VIRARBO file*

When VIRCONF is run with the PARM=UNLOAD parameter, the existing VIRARBO definitions are converted into control cards and are written to SYSPCH (VSE) or SYSPUNCH (MVS). The created cards issued by VIRCONF may be edited and then reused with another VIRCONF job with the PARM=LOAD parameter.

### 7.2.4. Verify control card syntax

Below are some JCL examples to verify the control card syntax:

```
* $$ JOB JNM=VIRCONF,CLASS=0,DISP=D
* $$ LST DISP=D,CLASS=V,DEST=(,SPTUSER)
// JOB VIRCONF SYNTAX CHECK
// LIBDEF *,SEARCH=(VIRT453.SUBLIB)
// EXEC VIRCONF,PARM='SCAN'
(insert sysin control statements here)
/*
/&
* $$ E0J
```

*VIRCONF JCL in VSE for syntax verification*

```
//VIRCONF JOB CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//* VIRCONF SYNTAX CHECK
//CONFCHK EXEC PGM=VIRCONF,PARM=SCAN
//STEPLIB DD DSN=yourqual.VIRT453.LOADLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
(insert sysin control statements here)
/*
```

*VIRCONF JCL in MVS for syntax verification*

Submitting the VIRCONF program with PARM=SCAN allows you to scan the SYSIPT (VSE) or SYSIN (MVS) cards for potential syntax errors. There is no access to the VIRCONF file.

### 7.2.5. Multi-language support

When uploading the VIRARBO file, VIRCONF may select one among several versions of a control card, based on the LANG=xx parameter defined in the JCL. In this way, the same SYSIN file may be used to generate several different language versions of the VIRARBO file. For example:

```
* $$ JOB JNM=VIRCONF,CLASS=0,DISP=D
// JOB VIRCONF LOAD VIRARBO
// LIBDEF *,SEARCH=(VIRT453.SUBLIB)
// DLBL VIRARBO,'VIRTEL.TESTARBO.KSDS',,VSAM,CAT=VSESPUC
// EXEC VIRCONF,PARM='LOAD,LANG=FR'
      TRANSACT ID=PC-0003,
      (FR) NAME='Entrée',
      (FR) DESC="Gestion des points d'entrée",
      (EN) NAME='Entry',
      (EN) DESC='Entry point management',
      APPL=VIR0044,
      TYPE=2,
      STARTUP=1
/*
/&
* $$ E0J
```

*VIRCONF JCL in VSE for multi-language upload*

```
//VIRCONF JOB CLASS=A,MSGCLASS=X,MSGLEVEL=(1,1),NOTIFY=&SYSUID
//* LOAD AN ARBO FILE USING MULTILINGUAL SOURCE
//VIRCONF EXEC PGM=VIRCONF,PARM='LOAD,LANG=EN'
//STEPLIB DD DSN=yourqual.VIRT453.LOADLIB,DISP=SHR
//SYSPRINT DD SYSOUT=*
//VIRARBO DD DSN=VIRTEL.TEST.ARBO,DISP=SHR
//SYSIN DD *
      TRANSACT ID=PC-0003,
      (FR) NAME='Entrée',
      (FR) DESC="Gestion des points d'entrée",
      (EN) NAME='Entry',
      (EN) DESC='Entry point management',
      APPL=VIR0044,
```



```

        TYPE=2,
        STARTUP=1
/*

```

*VIRCONF JCL in MVS for multi-language upload*

### 7.3. VIRCONF Control Cards

#### 7.3.1. VIRCONF control card syntax

The control card syntax for VIRCONF is similar to the syntax for JCL.

- Each instruction begins on a new card
- One instruction consists of an “operation code”, followed by a blank space, followed by one or more parameters, followed by an optional comment
- The parameters use the keyword=value form and parameters are separated with a comma
- The parameters are ended by a blank character; anything after this first blank is treated as a comment.
- A card beginning with a “\*” is treated as a comment
- A completely blank card is treated as a comment
- A card beginning with two characters between parenthesis, for instance (EN), will be processed only if these two characters match the value of the LANG= parameter specified in the JCL PARM
- Instructions must be coded between columns 1 to 71. Column 72, if non-blank, means that the instruction continues on the next card
- If a parameter is terminated by a comma followed by a blank, the instruction continues at the first non-blank character of next card. A non-blank in column 72 is optional in this case
- A character string between apostrophes or quotes which goes over column 71 may be continued on the next card, by putting a non-blank character in column 72 and by continuing the string starting on column 16 of the next card
- Each instruction must have at least one “ID=” parameter which is used as a key to identify the described entity
- The values of parameters may in general contain letters (A-Z in upper case), digits (0-9), and special characters (.+&\$\*-/\_%\_?:@). Some parameters also allow values which contain other special characters, letters in lower case, and blank characters, and in this case the value must be coded as a character string enclosed in either quotes or apostrophes

You can generate some examples by submitting a job using the PARM=UNLOAD parameter (see “[Unloading a VIRARBO file](#)”, page 87) for a specific VIRARBO file, for instance the one delivered as VIRARBO base in the standard installation process.

#### 7.3.2. APPLIC

This operation adds or replaces an APPLIC entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Applications Management” on page 123

Parameter	Item	Remarks
ID=	Name	
DESC=	Description	Quotes allowed
LOGON=	Logon	Quotes allowed
STATUS=	Status	Quotes allowed

### 7.3.3. DELETE

This operation deletes an entity of the specified type from the VIRARBO file.

Parameter	Item	Remarks
TYPE=	Entity type	LINE, RULE, TERMINAL, etc
ID=	Entity name	Quotes allowed

### 7.3.4. DEPT

This operation adds or replaces a DEPT entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Create a department” or “Profile lent to a department”.

Parameter	Item	Remarks
ID=	Department	
DESC=	Description	Quotes allowed
OWNER=	Responsible	
PROFILE=	Lent profiles list	Separated by commas, the whole between parenthesis

### 7.3.5. ENTRY

This operation adds or replaces an ENTRY entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Parameters of the Entry Point”.

Parameter	Item	Remarks
ID=	Name	
DESC=	Description	Quotes allowed
TRANSACT=	Transactions	
ARBO=	Arborescence	
ENDPAGE=	Last Page	
TRANSP=	Transparency	
TIMEOUT=	Time out	Numeric
EMUL=	Emulation	
SCENDIR=	Directory for scenarios	
SIGNON=	Signon program	
MENU=	Menu program	
IDENT=	Identification program or Identification scénario	
COMPR3=	Type 3 compression	
IDREQ=	Mandatory Identification	
SWAP=	3270 swap key	
EXTCOLOR=	Extended colors	

### 7.3.6. INDEX

This operation adds or replaces an INDEX entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Updating the keywords”.

Parameter	Item	Remarks
ID=	Keyword	Quotes allowed
TARGET=	Target Node - Choice	[1] ; Quotes allowed First 6 characters : Target Node Characters 7,8 : Choice

**Note [1]**

| If the target node name has less than 6 characters, it must be padded with blanks and enclosed in quotes.

### 7.3.7. LINE

This operation adds or replaces a LINE entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Parameters of the line”.

Parameter	Item	Remarks
ID=	Internal name	
NAME=	External name	
PARTNER=	Remote Ident	
LOCADDR=	Local ident	
DESC=	Description	Quotes allowed
TERMINAL=	Prefix	
ENTRY=	Entry Point	
TYPE=	Line type	
INOUT=	Possible calls	
COND=	Startup prerequisite	Quotes allowed
PROTOCOL=	Protocol program	
SECURITY=	Security program	
TIMEOUT=	Time out	Numeric
ACTION=	Action if T/O	Numeric
WINSZ=	Window	Numeric
PKTSZ=	Packet	Numeric
PAD=	Pad	
TRAN=	Tran	
RETRY=	Retries	Numeric
DELAY=	Delay	Numeric
UNIQUEP=	Unique partner [1]	Y or N
SHARED=	Shared address [1]	Y or N

**Note [1]**

| This parameter is available only in VIRCONF

### 7.3.8. NODE

This operation adds or replaces a NODE entity in the VIRARBO file. The parameters correspond to the various items described under the heading "Defining a native node".

Parameter	Item	Remarks
ID=	Name of Minitel node	Quotes allowed
PAGE=	Generic of associated pages	Quotes allowed
GUIDE=	Generic of associated guides	Quotes allowed
CHILD=	Generic of children	Quotes allowed

### 7.3.9. PC

This operation adds or replaces a PC entity in the VIRARBO file. The parameters correspond to the various items described under the heading "PC management".

Parameter	Item	Remarks
ID=	PC name	
DESC=	Description	Quotes allowed
DISABLE=	Block	X = blocked connections
SUBDIR=	Associated directory	
PASSCODE=	Password	Quotes allowed

### 7.3.10. PROFILE

This operation adds or replaces one entity with PROFILE entity in the VIRARBO file. The parameters correspond to the various items described under the heading "Defining a profile".

Parameter	Item	Remarks
ID=	Profil	
DESC=	Description	Quotes allowed
DEPT=	Department	
RESOURCE=	List of given resources	Separated by commas, and surrounded by parentheses

### 7.3.11. RESOURCE

This operation adds or replaces a RESOURCE entity in the VIRARBO file. The parameters correspond to the various items described under the heading "Defining a resource".

Parameter	Item	Remarks
ID=	Ressource	
DESC=	Description	Quotes allowed
DEPT=	Department	
TYPE=	Resource type	Unused field

### 7.3.12. RULE

This operation adds or replaces a RULE entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Parameters of the rule”.

Parameter	Item	Remarks
ID=	Rule name	
RULESET=	Ruleset name	[2]
STATUS=	Status	ACTIVE or INACTIVE
DESC=	Description	Quotes allowed
ENTRY=	Entry Point	
PARAM=	Parameter	Quotes allowed
TRACE=	Trace	
IPADDR=	IP Subnet	[1]
NETMASK=	Mask	
HTTPHOST=	Host	[1] ; Quotes allowed
EMAIL=	eMail	[1] ; Quotes allowed
CALLER=	Calling DTE	[1] ; Numeric; or IP address
CALLED=	Called	[1] ; Numeric
CUD0=	CUD0 (Hexadecimal)	[1] ; Hexadecimal
USERDATA=	User data	[1] ; Quotes allowed
DAYS=	Days	[1] ; Seven characters 'X' or blank, defining the days from Sunday to Saturday
TIME=	Start time	[1] ; Twelve numeric characters with this pattern HHMMSSHHMMSS

#### Note [1]

The conditions are coded in the form keyword=(condition,value) where condition is one of the following:

##### IGNORE

| Ignore

##### EQUAL

| Is

##### NOTEQ

| Is not

##### BEGIN

| Starts with

##### NOTBEGIN

| Does not

##### END

| Ends with

##### NOTEND

| Does not

#### Note [2]

| For compatibility with earlier versions, VIRCONF also accepts the parameter LINE= as a synonym of RULESET=

### 7.3.13. SERVER

This operation adds or replaces a SERVER entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Parameters of external server”.

Parameter	Item	Remarks
ID=	Name	
DESC=	Description	Quotes allowed
DIALNO=	Number	Numeric or '='
USERDATA=	Data	Quotes allowed
LINE=	Line number	
LINE2=	Backup line	
CALLER=	Caller	Numeric or '=' or '*'
EMUL=	Emulation	
CHARSET=	Character set	
TIMEOUT=	Server time out	Numeric
DELAY=	Inactivity delay	Numeric
ACTION=	Cut off warning	Numeric
LEVEL=	Price level	
SECRET=	Secret	
FACILITY=	Facilities	Hexadecimal or '='
CUD0=	CUD0 (hex)	Hexadecimal
TIOA=	TIOA at start up	Quotes allowed

### 7.3.14. SSERV

This operation adds or replaces an SSERV entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Defining a sub-server node”.

Parameter	Item	Remarks
ID=	Sub-server node name	Quotes allowed
SERVER=	Sub-server name	
TRANSACT=	Transaction name	[1] ; Quotes allowed
CONTROL=	Control program	
TIOA=	TIOA	[1] ; Quotes allowed
TRANSLAT=	Translation type	
RESTRICT=	Node with reserved access	[2] ; Quotes allowed
PFRETOUR=	Return key	[3] ; Quotes allowed
PFGUIDE=	Guide key	[3] ; Quotes allowed
PFSOMMR=	Summary key	[3] ; Quotes allowed
PFSUITE=	Next key	[3] ; Quotes allowed
PFREPET=	Repeat key	[3] ; Quotes allowed
PFANNUL=	Cancel key	[3] ; Quotes allowed

Parameter	Item	Remarks
PFCORREC=	Correct key	[3] ; Quotes allowed

**Note [1]**

| Specify either TRANSACT or TIOA, but not both

**Note [2]**

| \* = Yes

**Note [3]**

The key name must be coded according to the table below:

Key name	Code	Key name	Code	Key name	Code
Enter	" ' "	PF12	' @ '	PF24	' < '
PF1	1	PF13	A	PA1	' % '
PF2	2	PF14	B	PA2	' > '
PF3	3	PF15	C	PA3	' , '
PF4	4	PF16	D	Clear	' _ '
PF5	5	PF17	E		
PF6	6	PF18	F		
PF7	7	PF19	G		
PF8	8	PF20	H		
PF9	9	PF21	I		
PF10	' : '	PF22	' ç '		
PF11	' # '	PF23	' - '		

### 7.3.15. SUBDIR

This operation adds or replaces a SUBDIR entity in the VIRARBO file. The parameters correspond to the various items described under the heading "Parameter of directory".

Parameter	Item	Remarks
ID=	Directory name	
DESC=	Description	Quotes allowed
FSTYPE=	Type	
DDNAME=	DD name	
KEY=	Keyword	
NAMELEN=	Size of names	
CASESEN=	Case	
AUTHUP=	Copy up	X = copy into the directory is allowed
AUTHDOWN=	Copy down	X = copy from the directory is allowed
AUTHDEL=	Delete	X = file deletion is allowed

### 7.3.16. TERMINAL

This operation adds or replaces a TERMINAL entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Parameter of the terminal”.

Parameter	Item	Remarks
ID=	Terminal	
RELAY=	Relay	Quotes allowed
POOL=	Pool name	
DESC=	Description	Quotes allowed
ENTRY=	Entry Point	
RELAY2=	2nd relay	Quotes allowed
TYPE=	Type	
COMPRESS=	Compression	
INOUT=	Possible calls	Numeric
STATS=	Write Stats to	
REPEAT=	Repeat	Numeric

### 7.3.17. TRANSACT

This operation adds or replaces a TRANSACT entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Parameter of the transaction”.

Parameter	Item	Remarks
ID=	Internal name	
NAME=	External name	Quotes allowed
DESC=	Description	Quotes allowed
APPL=	Application	Quotes allowed
ALIAS=	Alias	
PASSTCKT=	PassTicket	Numeric
RACFNAME=	Name (for PassTicket)	
TYPE=	Application type	Numeric
TERMINAL=	Pseudo-terminals	
LOGMODE=	Logmode	
STARTUP=	How started	Numeric
SECURITY=	Security	Numeric
TRANSL=	Translation(s) or H4W commands	
LOGMSG=	Logon message or Check URL Prefix	Quotes allowed
TIOASTA=	TIOA at logon	Quotes allowed
TIOAEND=	TIOA at logoff	Quotes allowed
EXITSTA=	Initial Scenario	



Parameter	Item	Remarks
EXITEND=	Final Scenario	
EXITMSGI=	Input Scenario	
EXITMSGO=	Output Scenario	

### 7.3.18. UPDATE

This operation updates one or more parameters of an entity in the VIRARBO file.

Parameter	Item	Remarks
TYPE=	Entity type	LINE, RULE, TERMINAL, etc
ID=	Entity name	Quotes allowed
param=	According to the entity type	See the preceding description of the entity

### 7.3.19. USER

This operation adds or replaces a USER entity in the VIRARBO file. The parameters correspond to the various items described under the heading “Managing users”.

Parameter	Item	Remarks
ID=	User name	
NAME=	Description	Quotes allowed
DEPT=	Department	
PASSWORD=	Password (clear text)	[1]
PASSCODE=	Password (encrypted)	[1] ; Quotes allowed
ADMIN=	Administrator	1=Yes, 0=No
ASSIST=	Assist.(O-N)	1=Yes, 0=No
PROFILE=	List of profiles assigned	List of profiles assigned

#### Note [1]

Indicate either PASSWORD (readable password), or PASSCODE (encrypted password). PASSWORD allows you to specify the password explicitly; if present, it must be coded after the ID parameter. PASSCODE is generated by the UNLOAD function; this allows to unload and reload the USER records without exposing the password in readable format.

#### Note [2]

The PARM='UNLOAD,PLAINTEXT' function unloads the password in readable format (PASSWORD instead of PASSCODE). To use this function, the user that submits this VIRCONF job must be RACF authorized (or by another security tool) with READ access for the VIRTEL.PASSWORD.DECRYPT resource in the FACILITY class.

# 8. Security

## 8.1. How To Activate RACF Security

Perform the following steps to activate RACF security for VIRTEL in the MVS environment.

### 8.1.1. Modify the VIRTCT

In the VIRTCTxx member of the VIRTEL CNTL library,

- VIRTCTUS = English language.
- VIRTCTFR = French language.

replace the default parameters :

```
SECUR=NO , RAPPL=VIRTSEV , RNODE=VIRTNODE
```

with the following parameters:

```
SECUR=(RACROUTE , RACF) ,  
RAPPL=FACILITY , RNODE=FACILITY , PRFSECU=VIRTEL ,
```

This tells VIRTEL that its security definitions are stored in the FACILITY class, and the resource names are prefixed by “VIRTEL.”. You can choose your own prefix for each VIRTEL. Multiple VIRTEL started tasks can share the same resource name prefix if their security definitions are identical. You can also choose the class name, but it must already be defined in RACF with the correct attributes. It is recommended to use the FACILITY class which is standard in RACF.

Having updated the VIRTCTxx source member, reassemble and relink the VIRTCT into VIRTEL LOADLIB using the sample JCL in member ASMTCT of the VIRTEL CNTL library. Be sure to specify the correct member name MEMBER=VIRTCTxx in the job. Stop and start VIRTEL to pick up the new VIRTCT.

### 8.1.2. Add RACF definitions

The following RACF definitions are the minimum you need to get started. They simply authorize the VIRTEL administrator (you) to do everything. In this job, replace youruserid by the administrator’s RACF userid or group name. This JCL can be found in member RACFDEF in the VIRTEL SAMPLIB.

```
//VIRTRACF JOB 1 , RACFDEF , MSGCLASS=X , CLASS=A , NOTIFY=&SYSUID  
//*-----*  
//* RACF : AUTHORIZATIONS FOR VIRTEL *
```

```

/** Replace 'youruserid' by your VIRTEL administrator id */
/**-----*/
//STEP1 EXEC PGM=IKJEFT1A,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
/*-----*/
/* BY DEFAULT DISALLOW EVERYTHING TO GENERAL USERS */
/* BUT ALLOW EVERYTHING TO youruserid */
/*-----*/
RDEF FACILITY VIRTEL.* UACC(NONE)
PE VIRTEL.* CL(FACILITY) RESET ACC(READ) ID(youruserid)
/*-----*/
/* ALLOW EVERYONE TO USE THE 3270 LOGOFF TRANSACTION */
/*-----*/
RDEF FACILITY VIRTEL.PC-0020 UACC(READ) /* LOGOFF */
/*-----*/
/* REFRESH THE RACF PROFILES */
/*-----*/
SETR REFRESH RACLIST(FACILITY)
//

```

*RACFDEF : JCL to add RACF definitions*

### 8.1.3. Refine RACF definitions

Later you can refine the definitions so that other VIRTEL users can use VIRTEL transactions (such as secured VIRTEL Web Access transactions). The following example allows DEMOGRP to use transaction W2H-10:

```

//VIRTRACF JOB 1,RACFDEF,MSGCLASS=X,CLASS=A,NOTIFY=&SYSUID
/**-----*/
/** RACF : AUTHORIZATIONS FOR VIRTEL */
/**-----*/
//STEP1 EXEC PGM=IKJEFT1A,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*
//SYSTSIN DD *
/*-----*/
/* ALLOW DEMOGRP TO USE THE W2H-10 (CICS) TRANSACTION */
/*-----*/
RDEF FACILITY VIRTEL.W2H-10 UACC(NONE) /* CICS */
PE VIRTEL.W2H-10 CL(FACILITY) ACC(READ) ID(DEMOGRP)
/*-----*/
/* REFRESH THE RACF PROFILES */
/*-----*/
SETR REFRESH RACLIST(FACILITY)
//

```

*RACFDEF : JCL to update RACF definitions*

For more information about protecting VIRTEL Web Access resources, refer to the “Security” section of the VIRTEL Web Access Guide.

## 8.2. How To Activate ACF2 Security

Perform the following steps to activate ACF2 security for VIRTEL in the MVS environment.

### 8.2.1. Modify the VIRTCT

In the VIRTCTxx member of the VIRTEL CNTL library, replace the default parameters:

```
SECUR=NO,RAPPL=VIRTSEVR,RNODE=VIRTNODE
```

with the following parameters:

```
SECUR=RACROUTE,  
RAPPL=VIRTAPPL,RNODE=VIRTNODE,
```

This tells VIRTEL that the security definitions for calls to external servers are stored in the VIRTAPPL resource class, and that the security definitions for access to VIRTEL transactions, directories, and Minitel nodes are stored in the VIRTNODE resource class. You can choose your own resource class names for each VIRTEL. Multiple VIRTEL started tasks can share the same resource class names if their security definitions are identical.

Having updated the VIRTCTxx source member, reassemble and relink the VIRTCT into VIRTEL LOADLIB using the sample JCL in member ASMTCT of the VIRTEL CNTL library. Stop and start VIRTEL to pick up the new VIRTCT.

### 8.2.2. Determine the ACF2 resource type

ACF2 maps each 8-character SAF resource class name to a 3-character ACF2 resource type. By default, the resource type is the first three characters of the resource class name, so classes VIRTAPPL and VIRTNODE both map to resource type VIR. You can use the ACF2 CLASMAP record to translate the resource classes to different resource types if required.

### 8.2.3. Add ACF2 definitions

An example job to add VIRTEL definitions for ACF2 can be found in member ACF2DEF in the VIRTEL SAMPLIB. The commands in this job are explained in the following paragraphs.

#### 8.2.3.1. Create OMVS segment for VIRTEL

```
SET PROFILE(VIRTSTC) DIV(OMVS)  
INSERT VIRTSTC UID(0) HOME('/') PROGRAM('/bin/sh')
```

*ACF2DEF : ACF2 commands to create OMVS segment for VIRTEL*

This command allows VIRTEL to access the TCP/IP stack.

#### 8.2.3.2. Add permissions for VIRTEL administrators

```
$KEY(***** ) TYPE(VIR) UID(***** admin-group-name) SERVICE(READ)
```

*ACF2DEF : ACF2 command to grant administrator permissions*

This command permits users in group admin-group-name to access all VIRTEL transactions and administrator functions.

#### 8.2.3.3. Add permissions for VIRTEL general users

```
$KEY(W2H-10) TYPE(VIR) UID(***** user-group-name) SERVICE(READ)  
$KEY(CLI-****) TYPE(VIR) UID(***** user-group-name) SERVICE(READ)
```

*ACF2DEF : ACF2 commands to grant general user permissions*

These commands permit users in group user-group-name to access specific VIRTEL transactions.

Resource W2H-10 permits specific access to the CICS Web Access transaction on port 41001. Resource CLI-\*\*\*\* is a generic resource which permits access to customer-defined transactions (internal name CLI-nn) on port 41002 and to the directory CLI-DIR.

#### 8.2.3.4. Allow everyone to use the 3270 LOGOFF transaction

```
$KEY(PC-0020) TYPE(VIR) UID(***** *) SERVICE(READ)
```

*ACF2DEF : ACF2 command to permit access to 3270 Logoff transaction*

This command permits all users to use the 3270 Logoff transaction, whose internal name is PC-0020.

### 8.3. How To Activate Top Secret (TSS) Security

Perform the following steps to activate TSS security for VIRTEL in the MVS environment.

#### 8.3.1. Modify the VIRTCT

In the VIRTCTxx member of the VIRTEL CNTL library, replace the default parameters:

```
SECUR=NO ,RAPPL=VIRTSEVR ,RNODE=VIRTNODE
```

with the following parameters:

```
SECUR=(RACROUTE ,TOPS) ,  
RAPPL=VIRTAPPL ,RNODE=VIRTNODE ,
```

This tells VIRTEL that the security definitions for calls to external servers are stored in the VIRTAPPL resource class, and that the security definitions for access to VIRTEL transactions, directories, and Minitel nodes are stored in the VIRTNODE resource class. You can choose your own resource class names for each VIRTEL. Multiple VIRTEL started tasks can share the same resource class names if their security definitions are identical.

Having updated the VIRTCTxx source member, reassemble and relink the VIRTCT into VIRTEL LOADLIB using the sample JCL in member ASMTCT of the VIRTEL CNTL library. Stop and start VIRTEL to pick up the new VIRTCT.

#### 8.3.2. Add TSS definitions

A example job to add VIRTEL definitions for TSS can be found in member TOPSDEF in the VIRTEL SAMPLIB. The commands in this job are explained in the following paragraphs.

##### 8.3.2.1. Create VIRTEL facility

```
TSS MODIFY (FACILITY(USERnn=NAME=VIRTFAC))  
TSS MODIFY (FACILITY(VIRTFAC=PGM=VIR))  
TSS MODIFY (FACILITY(VIRTFAC=ACTIVE))  
TSS MODIFY (FACILITY(VIRTFAC=ASUBM))  
TSS MODIFY (FACILITY(VIRTFAC=AUTHINIT))  
TSS MODIFY (FACILITY(VIRTFAC=DEFACID(*NONE*)))  
TSS MODIFY (FACILITY(VIRTFAC=LUMSG))  
TSS MODIFY (FACILITY(VIRTFAC=MODE=FAIL))  
TSS MODIFY (FACILITY(VIRTFAC=MULTIUSER))
```

```
TSS MODIFY (FACILITY(VIRTFAC=NOABEND))
TSS MODIFY (FACILITY(VIRTFAC=NOAUDIT))
TSS MODIFY (FACILITY(VIRTFAC=NOPROMPT))
TSS MODIFY (FACILITY(VIRTFAC=NORES))
TSS MODIFY (FACILITY(VIRTFAC=NOTSOC))
TSS MODIFY (FACILITY(VIRTFAC=NOXDEF))
TSS MODIFY (FACILITY(VIRTFAC=RNDPW))
TSS MODIFY (FACILITY(VIRTFAC=SHRPRF))
TSS MODIFY (FACILITY(VIRTFAC=SIGN(M)))
TSS MODIFY (FACILITY(VIRTFAC=STMSG))
TSS MODIFY (FACILITY(VIRTFAC=WARNPW))
```

*TOPSDEF : TSS commands to create VIRTEL facility*

VIRTFAC is the VIRTEL facility name. You may choose your own name, but you must replace VIRTFAC in all of the following commands by the name you chose. Replace USERnn by the name of an unused user facility (for example, USER34).

---

### 8.3.2.2. Create VIRTEL division and department

```
TSS CREATE(VIRTDIV) NAME('VIRTEL DIVISION') TYPE(DIVISION)
TSS CREATE(VIRTDEP) NAME('VIRTEL DEPT') TYPE(DEPARTMENT) +
DIVISION(VIRTDIV)
```

*TOPSDEF : TSS commands to create VIRTEL division and department*

A division and department are created to contain the VIRTEL resources. You can choose your own names, or you can use an existing division and department. If you choose to use different names then the following commands must be modified accordingly.

---

### 8.3.2.3. Create ACID for the VIRTEL started task

```
TSS CREATE(VIRTSTC) NAME('VIRTEL STC') TYPE(USER) +
FAC(BATCH,STC) PASSWORD(NOPW,0) DEPARTMENT(VIRTDEP) +
MASTFAC(VIRTFAC) NODSNCHK NOVOLCHK
```

*TOPSDEF : TSS commands to create ACID for VIRTEL started task*

An ACID named VIRTSTC is defined in the BATCH and STC facilities to allow VIRTEL to execute in both batch and started task modes. It has no password and it belongs to department VIRTDEP.

The definition in the BATCH facility is not compulsory and is only required if VIRTEL might be executed as a batch job.

---

### 8.3.2.4. Assign VIRTEL procedure name to the ACID

```
TSS ADDTO(STC) PROCNAME(VIRTEL) ACID(VIRTSTC)
```

*TOPSDEF : TSS commands to associate ACID with VIRTEL started task*

This command associates the VIRTEL started task with the VIRTSTC ACID. VIRTEL is the name of the started task procedure in the system or user PROCLIB.

### 8.3.2.5. Create OMVS segment for VIRTEL

```
TSS ADDTO(VIRTSTC) UID(0) DFLTGRP(OMVSGRP) GROUP(OMVSGRP) +
OMVSPGM('/bin/sh') HOME('/')
```

*TOPSDEF : TSS commands to create OMVS segment for VIRTEL*

This command allows VIRTEL to access the TCP/IP stack. The name of the group (OMVSGRP in this example) should be adapted according to your naming conventions.

### 8.3.2.6. Define VIRTEL resource classes in the RDT

```
TSS ADDTO(RDT) RESCLASS(VIRTAPPL)
TSS ADDTO(RDT) RESCLASS(VIRTNODE)
```

*TOPSDEF : TSS commands to define VIRTEL resource classes*

VIRTEL uses two resource classes for security management.

The first class, whose name must match the RAPPL parameter of the VIRTCT, contains the names of VTAM applications used by VIRTEL Multi-Session, and the names of external servers used by VIRTEL Outgoing Calls (Videotex).

The second class, whose name must match the RNODE parameter of the VIRTCT, contains node names for VIRTEL Incoming Calls (Minitel), the names of sub-applications and directories for VIRTEL administration, and the internal names of transactions associated with entry points for VIRTEL Web Access.

### 8.3.2.7. Attach resources to VIRTEL department

TSS ADDTO(VIRTDEP) VIRTAPPL(AE)	Annuaire électronique
TSS ADDTO(VIRTDEP) VIRTAPPL(SNCF)	Serveur SNCF
TSS ADDTO(VIRTDEP) VIRTAPPL(\$\$ALLSRV)	Authorize all servers
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$ARBO\$)	Arborescence (admin.)
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$UTIL\$)	Users
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$APPL\$)	Applications
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$CMP3\$)	Compression
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$GLOG\$)	Entry points
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$LINE\$)	Lines
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$PCPC\$)	Intelligent terminals
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$RESO\$)	Network management
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$SECU\$)	Virtel security
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$SERV\$)	External servers
TSS ADDTO(VIRTDEP) VIRTNODE(\$\$TERM\$)	Terminals
TSS ADDTO(VIRTDEP) VIRTNODE(PC)	Administration transactions
TSS ADDTO(VIRTDEP) VIRTNODE(PC-0020)	Logoff transaction
TSS ADDTO(VIRTDEP) VIRTNODE(SERVEUR)	Transaction SERVEUR
TSS ADDTO(VIRTDEP) VIRTNODE(W2H)	Web Access transactions
TSS ADDTO(VIRTDEP) VIRTNODE(W2H-10)	Web Access CICS transaction
TSS ADDTO(VIRTDEP) VIRTNODE(CLI)	Client transactions

*TOPSDEF : TSS commands to define VIRTEL resources*

AE and SNCF are examples of external servers defined for VIRTEL Outgoing Calls (Videotex). VIRTEL permits access to an external server if the user is authorized to the corresponding resource name. Users who are authorized to the resource named \$\$ALLSRV may access all servers.

The resources named \$\$xxxx\$ are used to grant access to the various VIRTEL administrator functions. Refer to the VIRTEL Connectivity Reference manual for more details.

The resource named PC is a generic resource which permits access to the VIRTEL administrator 3270 interface transactions, whose internal name is PC-nnnn. The resource PC-0020 permits specific access to the 3270 Logoff transaction.

Resource W2H is a generic resource which permits access to VIRTEL Web Access transactions (internal name W2H-nn) and to the directory W2H-DIR. The resource W2H-10 permits specific access to the CICS Web Access transaction.

Resource CLI is a generic resource which permits access to customer-defined transactions (internal name CLI-nn) and to the directory CLI-DIR.

---

#### 8.3.2.8. Create administrator profile

```
TSS CREATE(VIRTADP) NAME('VIRTEL ADMINISTRATOR') +
    TYPE(PROFILE) DEPARTMENT(VIRTDEP)
TSS ADDTO(VIRTADP) FACILITY(VIRTFAC)
TSS PERMIT(VIRTADP) VIRTAPPL(AE)
TSS PERMIT(VIRTADP) VIRTAPPL(SNCF)
TSS PERMIT(VIRTADP) VIRTAPPL($$ALLSRV)
TSS PERMIT(VIRTADP) VIRTNODE($$ARBO$$)
TSS PERMIT(VIRTADP) VIRTNODE($$UTIL$$)
TSS PERMIT(VIRTADP) VIRTNODE($$APPL$$)
TSS PERMIT(VIRTADP) VIRTNODE($$CMP3$$)
TSS PERMIT(VIRTADP) VIRTNODE($$GLOG$$)
TSS PERMIT(VIRTADP) VIRTNODE($$LINE$$)
TSS PERMIT(VIRTADP) VIRTNODE($$PCPC$$)
TSS PERMIT(VIRTADP) VIRTNODE($$RESO$$)
TSS PERMIT(VIRTADP) VIRTNODE($$SECU$$)
TSS PERMIT(VIRTADP) VIRTNODE($$SERV$$)
TSS PERMIT(VIRTADP) VIRTNODE($$TERM$$)
TSS PERMIT(VIRTADP) VIRTNODE(PC(G))
TSS PERMIT(VIRTADP) VIRTNODE(SERVEUR)
TSS PERMIT(VIRTADP) VIRTNODE(W2H(G))
TSS PERMIT(VIRTADP) VIRTNODE(CLI(G))
```

*TOPSDEF : TSS commands to create VIRTEL administrator profile*

The VIRTEL administrator profile is named VIRTADP. You may choose a different name if required. In this example the administrator is granted access to all of the VIRTEL administration functions as well as to transactions PC-nnnn, W2H-nn and CLI-nn, and to directories W2H-DIR and CLI-DIR.

---

#### 8.3.2.9. Create user profile

```
TSS CREATE(VIRTUSP) NAME('VIRTEL USER') +
    TYPE(PROFILE) DEPARTMENT(VIRTDEP)
TSS ADDTO(VIRTUSP) FACILITY(VIRTFAC)
TSS PERMIT(VIRTUSP) VIRTAPPL(AE)
TSS PERMIT(VIRTUSP) VIRTAPPL(SNCF)
TSS PERMIT(VIRTUSP) VIRTNODE(W2H-10)
```

*TOPSDEF : TSS commands to create VIRTEL user profile*

The VIRTEL general user profile is named VIRTUSP. You may choose a different name if required. In this example the general user is granted access to external servers AE and SNCF, as well as to transaction W2H-10.



### 8.3.2.10. Allow everyone to use the 3270 LOGOFF transaction

```
TSS PERMIT(ALL) VIRTNODE(PC-0020)
```

*TOPSDEF : TSS command to permit access to 3270 Logoff transaction*

This command permits all users to use the 3270 Logoff transaction, whose internal name is PC-0020.

### 8.3.2.11. Define VIRTEL general users

```
TSS ADDTO(userid1) PROFILE(VIRTUSP)  
TSS ADDTO(userid2) PROFILE(VIRTUSP)
```

*TOPSDEF : TSS command to add general users*

These commands define userid1 and userid2 as VIRTEL general users by adding the VIRTEL user profile to their ACID.

### 8.3.2.12. Define VIRTEL administrators

```
TSS ADDTO(admin1) PROFILE(VIRTADP)  
TSS ADDTO(admin2) PROFILE(VIRTADP)
```

*TOPSDEF : TSS command to add administrators*

These commands define admin1 and admin2 as VIRTEL administrators by adding the VIRTEL administrator profile to their ACID.

### 8.3.2.13. Authorize the VIRTEL LOADLIB

The VIRTEL load library should normally be APF-authorized. If this is not the case, NOAUTH should be specified in the VIRTFAC facility.

# 9. 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.

## 9.1. Open Source Software

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

jQuery

| Under MIT license  
| <https://jquery.org/license/>.

StoreJson

| Under MIT license  
| <https://github.com/marcuswestin/store.js/commit/baf3d41b7092f0bacd441b768a77650199c25fa7>.

jQuery\_UI

| Under MIT license  
| [http://en.wikipedia.org/wiki/JQuery\\_UI](http://en.wikipedia.org/wiki/JQuery_UI).

# Index

- ACCUEIL**
  - VIRTCT parameters , 48
- ACF2**
  - Security , 99
- ADDR1**
  - VIRTCT parameters , 49
- ADDR2**
  - VIRTCT parameters , 49
- AIC**
  - VIRTCT parameters , 49
- ANNUL**
  - VIRTCT parameters , 49
- APPLIC , 89**
- APPLID**
  - VIRTCT parameters , 49
- Applying to MVS**
  - PTF , 20
- Applying to VSE**
  - PTF , 41
- Applying updates , 21, 40**
- APPSTAT**
  - VIRTCT parameters , 50
- ARBO**
  - VIRTCT parameters , 50
- Assembly**
  - VIRTCT , 82
- BATCH1**
  - VIRTCT parameters , 50
- BATCH2**
  - VIRTCT parameters , 51
- BFVSAM**
  - VIRTCT parameters , 51
- BUFDATA**
  - VIRTCT parameters , 51
- BUFSIZE**
  - VIRTCT parameters , 51
- CHARSET**
  - VIRTCT parameters , 51
- CODE**
  - VIRTCT parameters , 52
- COMPANY**
  - VIRTCT parameters , 52
- COMPR3**
  - VIRTCT parameters , 52
- Configuration**
  - Trace , 75
- CORRECT**
  - VIRTCT parameters , 52
- COUNTRY**
  - VIRTCT parameters , 52
- CRYPT1**
  - VIRTCT parameters , 53
- CRYPT2**
  - VIRTCT parameters , 55
- DEFENTR**
  - VIRTCT parameters , 55
- DEFUTF8**
  - VIRTCT parameters , 55
- DELETE , 90**
- DEPT , 90**
- DIRECT**
  - VIRTCT parameters , 56
- DONTSWA**
  - VIRTCT parameters , 57
- ENTRY , 90**
- EXII1**
  - VIRTCT parameters , 57
- EXIT2**
  - VIRTCT parameters , 57
- EXIT3**
  - VIRTCT parameters , 57
- EXIT4**
  - VIRTCT parameters , 57
- EXIT5**
  - VIRTCT parameters , 57
- EXIT6**
  - VIRTCT parameters , 58
- EXIT7**
  - VIRTCT parameters , 58
- EXIT8**
  - VIRTCT parameters , 58
- EXPIRE**
  - VIRTCT parameters , 58
- FASTC**
  - VIRTCT parameters , 58
- FCAPT**
  - VIRTCT parameters , 58
- FCMP3**
  - VIRTCT parameters , 59
- File**
  - Statistics , 71
- GATE**
  - VIRTCT parameters , 59
- GMT**
  - VIRTCT parameters , 59
- GRNAME**
  - VIRTCT parameters , 60
- GTLOAD**
  - VIRTCT parameters , 60
- GTPRFE1**
  - VIRTCT parameters , 60
- GTPRFE2**
  - VIRTCT parameters , 60
- GTVSAM**
  - VIRTCT parameters , 61
- GTVSKIP**
  - VIRTCT parameters , 61
- GUIDE**
  - VIRTCT parameters , 61
- HTFORWD**
  - VIRTCT parameters , 61
- HTHEADR**
  - VIRTCT parameters , 61
- HTMINI**
  - VIRTCT parameters , 62
- HTPARM**
  - VIRTCT parameters , 62
- HTSETn**
  - VIRTCT parameters , 62
- HTVSAM**
  - VIRTCT parameters , 63
- IBERTEX**
  - VIRTCT parameters , 63
- IGNIDRFULL parameter , 20**
- IGNLU**
  - VIRTCT parameters , 63
- IND\$FILE , 62**
- iv-remote-address , 61**
- LANG**
  - VIRTCT parameters , 63
- LICENCE**
  - VIRTCT parameters , 64
- LINE , 91**
- LOCK**
  - VIRTCT parameters , 64
- MARK**
  - VIRTCT parameters , 64

- MAXAPPL**
  - VIRTCT parameters , 64
- MEMORY**
  - VIRTCT parameters , 64
- MINITEL**
  - VIRTCT parameters , 65
- MQ1**
  - VIRTCT parameters , 65
- MQ2**
  - VIRTCT parameters , 65
- MULTI**
  - VIRTCT parameters , 65
- NBCVC**
  - VIRTCT parameters , 66
- NBDYNAM**
  - VIRTCT parameters , 66
- NBTERM**
  - VIRTCT parameters , 66
- NODE , 92**
- NUMTASK**
  - VIRTCT parameters , 66
- OSCORE**
  - VIRTCT parameters , 67
- OTMAPRM**
  - VIRTCT parameters , 66
- PACKET**
  - VIRTCT parameters , 67
- PASSTCK**
  - VIRTCT parameters , 67
- PassTicket , 67**
- Patch VIRTTEL program , 84**
- PC , 92**
- PREZ900**
  - VIRTCT parameters , 67
- PRFSECU**
  - VIRTCT parameters , 67
- PROFILE , 92**
- PWPROT**
  - VIRTCT parameters , 68
- RACAPPL**
  - VIRTCT parameters , 68
- RACF**
  - Security , 98
- RAPPL**
  - VIRTCT parameters , 68
- REALM**
  - VIRTCT parameters , 69
- REPET**
  - VIRTCT parameters , 69
- RESOURCE , 92**
- RESO**
  - VIRTCT parameters , 69
- RETOUR**
  - VIRTCT parameters , 69
- RNODE**
  - VIRTCT parameters , 70
- RULE , 93**
- SECUR**
  - VIRTCT parameters , 70
- SERVER , 94**
- SILENCE**
  - VIRTCT parameters , 71
- SNAPW**
  - VIRTCT parameters , 71
- SOMMR**
  - VIRTCT parameters , 71
- SSERV , 94**
- STATDSN**
  - VIRTCT parameters , 71
- STATS**
  - VIRTCT parameters , 71
- STEPLIB libraries**
  - ICSF , 22
  - MQSeries , 22
- STRNO**
  - VIRTCT parameters , 72
- SUBDIR , 95**
- SUITE**
  - VIRTCT parameters , 72
- SWAP**
  - VIRTCT parameters , 73
- SYSCONE , 73**
- SYSPLUS**
  - VIRTCT parameters , 73
- TCP1**
  - VIRTCT parameters , 73
- TCP2**
  - VIRTCT parameters , 74
- TERMINAL , 96**
- TIMEOUT**
  - VIRTCT parameters , 74
- TIMERQS**
  - VIRTCT parameters , 74
- TITRE1**
  - VIRTCT parameters , 74
- TITRE2**
  - VIRTCT parameters , 75
- Top Secret**
  - Security , 101
- TRACALL**
  - VIRTCT parameters , 75
- TRACBIG**
  - VIRTCT parameters , 75
- TRACEB**
  - VIRTCT parameters , 75
- TRACEOJ**
  - VIRTCT parameters , 75
- TRACEON**
  - VIRTCT parameters , 76
- TRACTIM**
  - VIRTCT parameters , 76
- TRANSACT , 96**
- TRAN**
  - VIRTCT parameters , 76
- UFILEn**
  - VIRTCT parameters , 76
- UPDATE , 97**
- USER , 97**
- VIRCONF , 85**
- Virconf**
  - LOAD , 85
  - NOREPL , 87
  - PLAINTXT , 97
  - SCAN , 88
  - UNLOAD , 87
- VIRSECU**
  - VIRTCT parameters , 77
- VIRSV1**
  - VIRTCT parameters , 77
- VIRTCT01 , 48**
- VSAM**
  - Sharing files , 80
- VSAMTYP**
  - VIRTCT parameters , 77
- VTAM application relay , 42**
- VTAM Primary ACB , 42**
- VTKEYS**
  - VIRTCT parameters , 78
- VTOWER**
  - VIRTCT parameters , 78
- X-Forwarded-For , 61**
- XM1**
  - VIRTCT parameters , 78
- XM2**
  - VIRTCT parameters , 78
- XMIT file , 16**
- ZAPH**
  - VIRTCT parameters , 79