

21CS VSEⁿ and z/VM Synergies

Edmund Wilhelm
VSEⁿ Core Senior Software Engineer



Agenda



- [Introduction](#)
- [Terminal Emulation Setup](#)
- [VSEⁿ Console Setup Prior to IPL](#)
- [Operating Environments](#)
- [Initial Installation Considerations](#)
- [IBM z/VM VSEⁿ Interface Routines](#)
- [Linux Fast Path \(LFP\)](#)
- [REXX Portability](#)
- [DASD Sharing](#)
- [Important VM/CP Functions](#)



Introduction



VSEⁿ can run as a guest system under any supported IBM z/VM® release.

In an environment where VSEⁿ runs under VM, you can:

- Log on to VM using the ID and password defined for the VSEⁿ virtual machine. Your terminal can then serve as the VSEⁿ console.
- Dial into the VSEⁿ system and use the functions provided by the Interactive User Interface.
- Log on to CMS and interact with VSEⁿ. VSEⁿ provides the VM/VSEⁿ Interface, which allows CMS users of VM to work with VSEⁿ systems concurrently.



Terminal Emulation Setup

Telnet3270

Host Definition Security Setup Printer Association

	Host Name or IP Address	LU or Pool Name	Port Number
Primary	192.168.x.x		23
Backup 1			23
Backup 2			23

Connection Options

Connection Timeout: 6 Seconds

Auto-reconnect

Try connecting to last configured host infinitely

Keep Alive

Enable Telnet Keep Alive

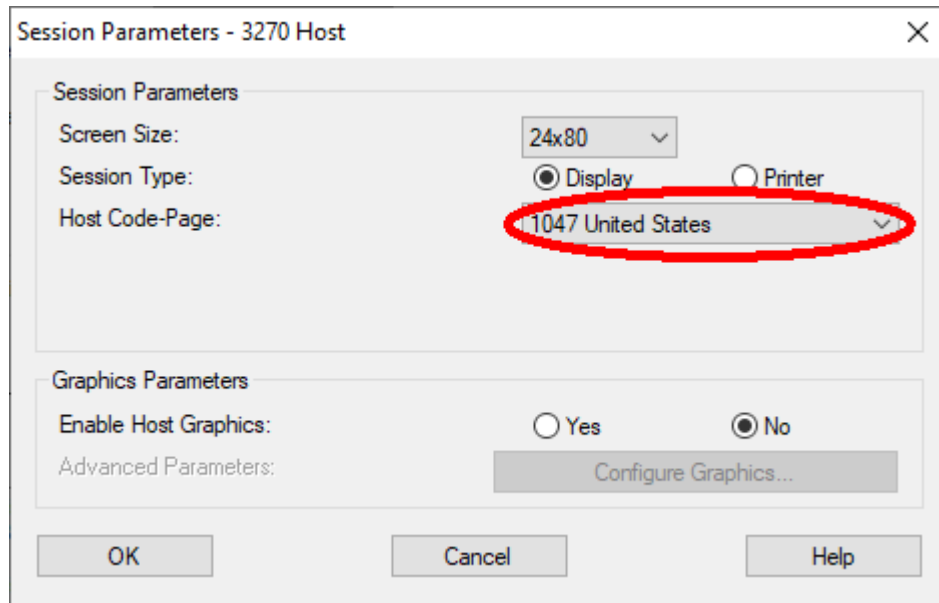
Keep Alive Time Out: 180 Seconds

OK Cancel Apply Help

- If the session is disconnected from the host, and if this box is checked, you will be reconnected automatically.
- The default is "not checked".



Terminal Emulation Setup cont.



- IBM z/VSE based on Host Code-page 037
- 21CS standardized on Host Code-page 1047
- Navigator has been using 1047 for years
- C for VSEⁿ 1.2 interprets the source code in code page 1047 by default.
- IBM APAR PH5019 (21CS VA00081) for HLASM for correct ASCII/Unicode data



Terminal Emulation Setup cont.

Two important file transfer definitions

The screenshot shows the 'File Transfer Settings' dialog box with the 'VM' tab selected. The 'TransferType' dropdown is set to 'binary80'. Under 'Transfer Options', the 'ascii', 'crif', and 'append' checkboxes are unchecked. The 'Record Format' dropdown is set to 'Fixed', and the 'Logical Record Length' text box contains the value '80'. The 'Additional Options' text box is empty. At the bottom, the 'OK' button is highlighted.

The screenshot shows the 'File Transfer Settings' dialog box with the 'VM' tab selected. The 'TransferType' dropdown is set to 'dump'. Under 'Transfer Options', the 'ascii', 'crif', and 'append' checkboxes are unchecked. The 'Record Format' dropdown is set to 'Fixed', and the 'Logical Record Length' text box contains the value '4112'. The 'Additional Options' text box is empty. At the bottom, the 'OK' button is highlighted.



VSEⁿ Console Setup Prior to IPL

- SET RUN ON
- SP RDR CONT EOF CLASS A
- DEFINE STORAGE 4G
- Virtual machine runs while a CP-initiated read is outstanding at your terminal.
- Spool continues virtual reader with a unit exception on the first read past the end of the file. Further reads result in a normal read of the next file. Only process class A files.
- Use DEFINE STORAGE to change the storage size of your virtual machine



VSEⁿ Console Setup Prior to IPL



- `TERMINAL CONMODE 3270`
- `TERMINAL BREAKIN MINIMAL`
- Defines virtual I/O support for 3270 with CCW processing.
- Specify that only high-priority warnings from CP and messages issued synchronously during `DIAGNOSE X'08'` processing will break in.



Operating Environments



Note: Irrespective of which IBM z/VM® version you are using, you are strongly recommended to define sufficient storage in the virtual guest machine so that your VSEⁿ will run without paging (that is, as a NOPDS system). As a result, only IBM z/VM® will perform paging.

Since VSEⁿ can run as a guest system under IBM z/VM® Version 5 Release 4 (or later), *the only VM mode that is available is:*

V=V (Virtual=Virtual)



Operating Environments cont.

VM Mode V=V

If VSEⁿ runs as a V=V guest, double paging (VM and VSEⁿ) will occur if VSEⁿ has a page data set (PDS).

System Function	Support
VM Paging	YES
VSE ⁿ Paging	
PDS (Page Data Set) – VSE ⁿ Paging	YES
NOPDS (No Page Data Set) – VSE ⁿ Paging	NO
VM CCW Translation	YES
VSE ⁿ CCW Translation	YES
VSE ⁿ Real Size	≥32MB (real from DEF STOR)
VSE ⁿ Virtual Size	xxGB (virtual from VSIZE definition)



Initial Installation Considerations



VSEⁿ uses device sensing to automatically define:

- Each device attached to VSEⁿ.
- Each device defined in the VM directory entry for VSEⁿ.

You should attach the devices to be used by VSEⁿ *before* the sense IPL. Those devices are automatically added to the IPL procedure and to the hardware configuration dialog. If you do not, you must later modify the IPL procedure created.

Detach devices you do *not* want in the VSEⁿ IPL procedure (for example, CMS minidisks or MAINT's 190 minidisk) or delete them later via the *Hardware Configuration* dialog.



IBM z/VM VSEⁿ Interface Routines



Virtual Machine Communication Facility (VMCF) Console

When running under VM, you also can have a VMCF console. In order to work with the VMCF console you must have installed the VM-VSE support on the VM userID from which you want to use the VMCF console function. Use member SKVMVSE in ICCF library 59 to install this support. After doing this you can use 'VSECMD' to send any VSE command to the VSE console. The command responses are routed back to the userID from where the command was issued.



IBM z/VM VSEⁿ Interface Routines cont.

- VSEⁿ system should be IPLed with SYS option VMCF=YES.
- The default value is VMCF=YES if VSEⁿ is a VM guest and VMCF=NO for native VSEⁿ.
- The option is ignored if VSEⁿ does not run in a VM virtual machine.
- The VM/VSEⁿ Interface is a set of VSEⁿ phases and CMS modules supplied by VSEⁿ.
- These phases and modules provide functions for interfacing to one or more VSEⁿ guest systems from CMS.
- The VM/VSEⁿ Interface routines are distributed in IJSYSRS.SYSLIB. Normally, you must obtain the routines from the library and install them on a CMS minidisk.



IBM z/VM VSEⁿ Interface Routines cont.



Functions Supported

1. Have none, some, or all messages from a job or from the system echoed to a specified owner (CMS user ID).
2. Reply to messages resulting from the execution of a job. The job must have a unique job owner ID (CMS user ID).
3. Submit jobs from a CMS terminal to a VSEⁿ guest system.
4. Issue VSEⁿ commands (including REDISPLAY commands) to a VSEⁿ guest system and have the resulting messages echoed to the CMS user.
5. Issue CP commands for execution in the virtual machine and have the resulting CP messages routed to the CMS job owner.



IBM z/VM VSEⁿ Interface Routines cont.

- VSEⁿ provides skeleton SKVMVSE in VSE/ICCF library 59. You use this skeleton to punch the MODULES, EXPLAINS, and EXECs from the VSEⁿ guest system to your VM machine.
- To ensure that the punch job works correctly, the VSE/POWER punch writer must be started with the VM parameter and must be enabled to process class A jobs:

```
PSTART PUN,00D,AB,,VM
```



IBM z/VM VSEⁿ Interface Routines cont.



Installation Tasks

1. First copy the skeleton SKVMVSE from VSE/ICCF library 59 to your primary library. You can use the Program Development Library dialog to copy VSE/ICCF library members.
2. Next, edit the copy of the skeleton. Replace the -V001- variable with the VSE/POWER destination parameter for punch output. This is the same name as the VM user ID where you want the VM/VSEⁿ Interface installed.

Note: When completing the copy of the skeleton, do not overwrite the comma, which is included in parentheses. It is not part of the variable -V001-.

When the job is run, the members are placed in the reader queue of the destination ID specified in the variable.



IBM z/VM VSEⁿ Interface Routines cont.

Installation Tasks cont.

3. Do not use the same user IDs in VSEⁿ and VM. If identical user IDs exist, you must make sure that they identify the same person. Identical user IDs cannot be used concurrently to access VSEⁿ console functions both from CMS and VSEⁿ in LPAR mode.
4. You should access the CMS minidisk where the routines will be loaded. The default is the first accessed R/W minidisk.
5. The VM/VSEⁿ Interface is activated during IPL (SYS command VMCF=YES).



IBM z/VM VSEⁿ Interface Routines cont.

Command Facilities of the VM/VSEⁿ Interface

- Execute VSE commands on a VSEⁿ virtual machine and have the resulting responses echoed to your VM user ID. You use the VSECMD command.

VSECMD also serves to

- Reply to messages that result from the execution of a job. The job must have a unique job owner ID (your VM user ID, for example).
- Issue CP commands for execution in the VSEⁿ virtual machine and have the resulting messages routed to your VM user ID.

Two facilities are available that let you issue CP commands:

- Enter CP commands (* CP command) from your VSEⁿ console
- Execute CP commands (// EXEC CPCMD) within JCL statements.



IBM z/VM VSEⁿ Interface Routines cont.



Command Facilities of the VM/VSEⁿ Interface cont.

- Submit jobs from a CMS terminal to a VSEⁿ virtual machine and have messages from the job echoed to a specific user ID (provided the ECHO/ECHOU parameter in the * \$\$ JOB statement had specified that user ID). You use the SUBVSE command.
- ECHOU routes most job messages ONLY to user ID. Reduces master console traffic. Useful for interactive debugging.



IBM z/VM VSEⁿ Interface Routines cont.



Command Authority

- If your system was IPLed with SEC=YES, command authority is established through the MCONS or AUTH parameters, within a user profile, of the VSEⁿ Access Control table DTSECTAB. You should have the following entry in your DTSECTAB in order to use the VM/VSEⁿ Interface:

```
*-----*  
*** CMS USER  
*-----*  
DTSECTAB TYPE=USER, C  
NAME=<VM UserID>, C  
PASSWRD=DUMMY, C  
READDIR=YES, C  
MCONS=YES, C  
AUTH=YES, C  
RIGHT=BTRANS
```



IBM z/VM VSEⁿ Interface Routines cont.



Console Messages to VM User ID

- You can have console messages sent to your VM User ID. You can have the messages from a particular JOB sent to your VM UserID via the ECHO parameter of the POWER JOB statement:

```
* $$ JOB JNM=TESTJOB, CLASS=4, DISP=D, ECHO= (ALL, <UserID>)
```

- You can also have all console messages sent to your VM UserID via the AR command **SYSECHO** (provided your VSE system was IPLed with SYS option VMCF=YES). This function allows to automate VSE operations from a CMS program.

```
SYSECHO ON, <UserID>
```



IBM z/VM VSEⁿ Interface Routines cont.



VSECMD: Send Commands and Replies to the VSE Virtual Machine

- **VSECMD** allows you to send commands to the VSEⁿ virtual machine for execution. These can be attention routine commands such as **MAP**, **PRTY**, **VSE/POWER** commands, **VCDD** commands etc.

```
VSECMD <vse_id> <command>
```



IBM z/VM VSEⁿ Interface Routines cont.



SIR VMCF

Syntax: SIR VMCF[={ON|OFF}]

This command allows to display the status of the VMCF interface and to alter it. Use the command to turn the VMCF interface on after it has been turned off inadvertently e.g., by a CMS user using VSECMD with incorrect authorization.

Message VMCF12I CMS-VSEⁿ CONSOLE INTERFACE DEACTIVATED will inform the operator that the interface has been turned off.

SIR VMCF=ON will restart the interface.



IBM z/VM VSEⁿ Interface Routines cont.



CORCMD You can use the CORCMD to retrieve console router internal information.

CORCMD STATUS=CONS

returns status info about the consoles known to the console router.

```
corcmd status=cons
```

```
Number of consoles: Act=00000003 Sus=00000000 Ud=00000001 Al=00000001
```

```
SYS 001BD3A4 A--- VMC 015DB891 A--- IC 015DB122 A--U
```

```
End of STATUS=CONS
```

Explanation:

Number of consoles: ACTIVE, SUSPENDED, accepts UD msgs, accepts ALERTS

Status: Active, Suspended, Netview(automa.), Undel. msgs



IBM z/VM VSEⁿ Interface Routines cont.

Explanation cont.:

Shows all consoles currently registered to the console router. Typically, you will see these consoles.

ConsName

SYS

VMC

IC

ConsName

Description

system console

VMCF console, only when running under VM

integrated console, only when supported by hardware. All z systems support the integrated console.

typically, the userID of a user, using the console function, or any other name as used by any console application (e.g., vendor products)



IBM z/VM VSEⁿ Interface Routines cont.



Console Hang

Sometimes a user claims that a console hangs. In this situation, first check whether the whole system is affected (hardwait, loop) or just the console.

If the system is still working, then try to get up a second console. Normally the easiest way is to logon to the IUI and use the console function.

Use the **CORCMD FORCE=consname** with the name of the affected console. If the console is the system console the console name is SYS. Entering this command normally frees the affected console immediately. Hitting the `enter` key at the console should reactivate it.



IBM z/VM VSEⁿ Interface Routines cont.

Console Hang cont.

If an IUI console is not available, any other console can be used. For example, you could use one of these consoles:

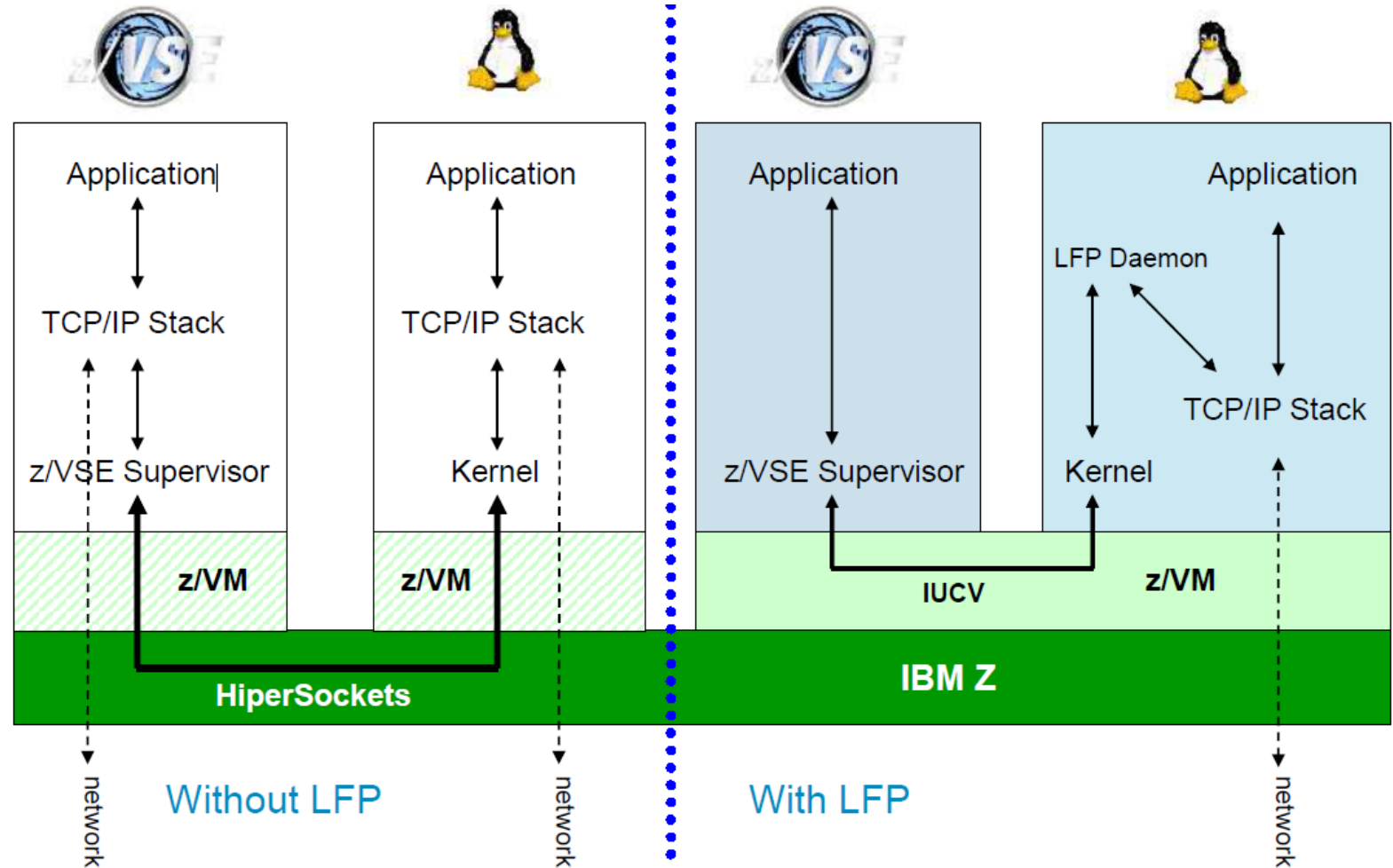
1. under VM, use the VSECMD to issue the CORCMD.
2. the integrated console, if the hardware supports it.
3. VM also emulates the integrated console. In CP mode enter VI
VMSG CORCMD ...



Linux Fast Path (LFP)

In z/VM-mode

Faster communication between VSEⁿ and Linux applications under z/VM

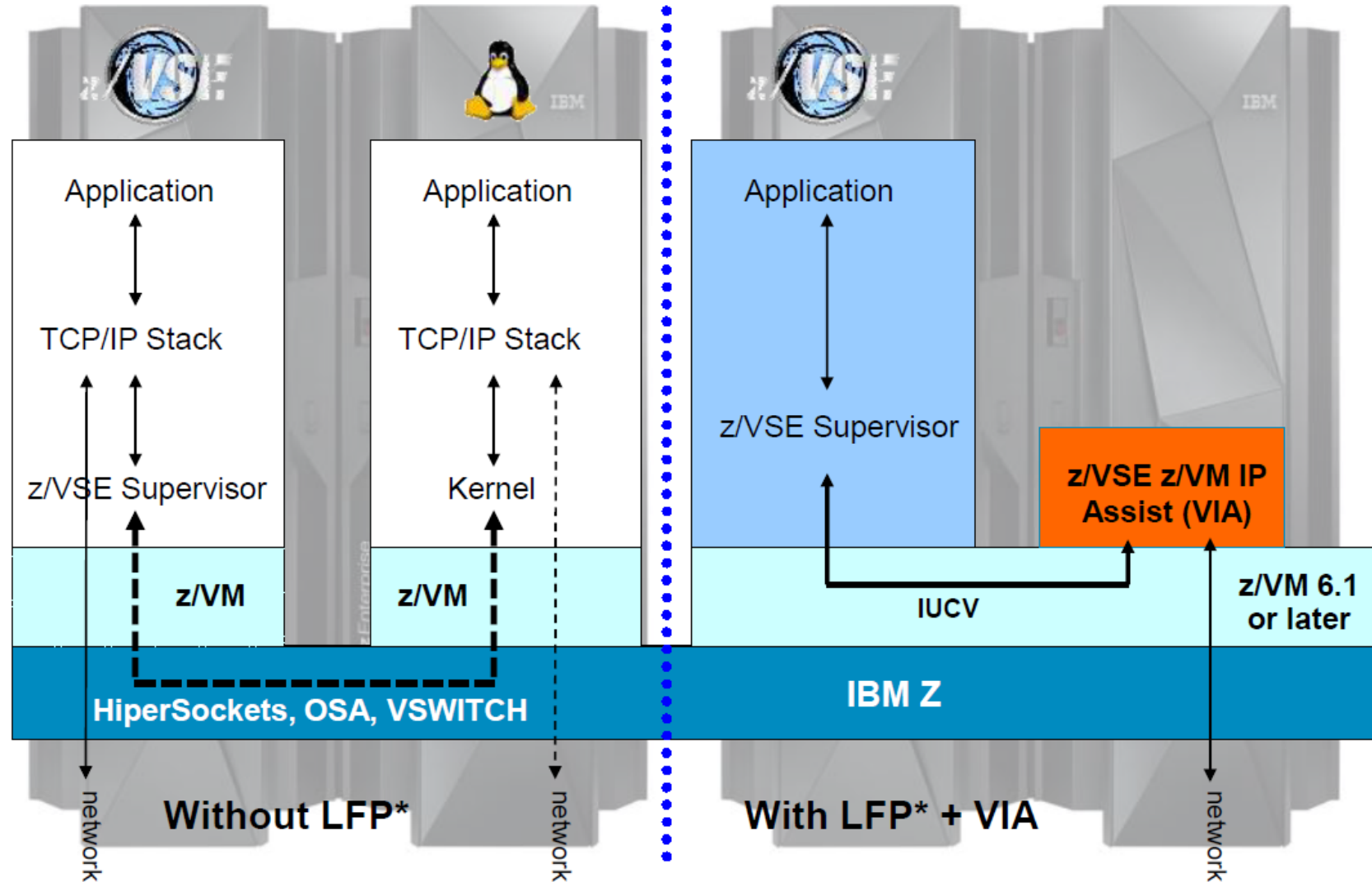




Linux Fast Path (LFP) cont.

VSEⁿ with z/VM IP Assist (VIA)

With z/VM IP Assist (VIA), no Linux is needed to utilize the LFP advantage





REXX Portability



A REXX program compiled under VM CMS can run under REXX/VSE.

Benefits of Using a Compiler

- Improved performance
- Reduced system load
- Protection for source code and programs
- Improved productivity and quality
- Portability of compiled programs
- SAA compliance checking



REXX Portability cont.



```
/* TITLE: FIBONACCI                                */
ARG limit repeat

DO I=1 TO repeat
  CALL FIBONACCI(limit)
  IF I = repeat THEN SAY 'Repeated' I 'times.'
END

SAY '0'
/* List all elements in the array */
DO K=1 TO limit
  SAY ARR.K
END

EXIT

FIBONACCI: PROCEDURE EXPOSE ARR.

CURRENT = ARG(1)

IF DATATYPE(ARR.CURRENT) = "NUM" THEN RETURN ARR.CURRENT

IF CURRENT = 1 | CURRENT = 2 THEN
  ARR.CURRENT = 1
ELSE
  ARR.CURRENT = FIBONACCI(CURRENT-2) + FIBONACCI(CURRENT-1)

RETURN ARR.CURRENT

/*
// JOB FIBONACC
// SETPARM USER='<USERID>'
// SETPARM PWD='<PASSWORD>'
// SETPARM LIMIT='249'
// SETPARM REPEAT='10000000'
// ID USER=&USER,PWD=&PWD
// LIBDEF PROC,SEARCH=(<LIB.SUBLIB>)
// EXEC REXX=FIBONACC,PARM='&LIMIT &REPEAT'
/*
/&
```



REXX Portability cont.



To compile:

- Upload to VM as text
- Compile with REXXC / (NOCE COMPILE CONDENSE OBJECT XREF
- Download text deck as .OBJ fixed bin 80
- Copy .OBJ to <LIB.SUBLIB> (with Navigator)



REXX Portability cont.



- Submit REXXLINK.JOB:

```
// JOB REXXLINK
// SETPARM INLIB='<LIB.SUBLIB>'
// SETPARM INNAME='FIBONACC'
// SETPARM OUTLIB='PRD2.CONFIG'
// SETPARM OUTNAME='FIBONACC'
// LIBDEF PHASE,CATALOG=&OUTLIB
// LIBDEF PROC,SEARCH=(PRD1.BASE)
// EXEC PROC=REXXLINK,STUBNAM=VSE,INLIB='&INLIB',INNAME=&INNAME,OUTLIB=C
//          '&OUTLIB',OUTNAME=&OUTNAME

/*
/&
```



REXX Portability cont.



- Submit JOB to use Compiled Version (PHASE) :

```
// JOB FIBONACC
// SETPARM USER= '<USERID>'
// SETPARM PWD= '<PASSWORD>'
// SETPARM LIMIT='249'
// SETPARM REPEAT='10000000'
// ID USER=&USER, PWD=&PWD
// EXEC FIBONACC, PARM='&LIMIT &REPEAT'
/*
/&
```



REXX Portability cont.



Interpreted

EOJ FIBONACC MAX.RETURN CODE=0000 DATE 06/14/2023, CLOCK 10/59/28, DURATION 00/02/48

Compiled

EOJ FIBONACC MAX.RETURN CODE=0000 DATE 06/14/2023, CLOCK 12/32/36, DURATION 00/00/06



REXX Portability cont.



Restrictions

REXXIPT

REXXIPT lets a program (called with ADDRESS JCL, ADDRESS LINK or ADDRESS LINKPGM) read data stored in compound variables as if it were SYSIPT data.

To have access to SYSIPT data, you need to use the JCL card // EXEC REXX= to call the program that contains the REXXIPT function call.



DASD Sharing

- All VSE systems must share the same lock file
- All shared DASDs must be ADDED with the SHR option on all systems having access to the volume
- Under VM.
 - all shared MDISKS must be defined with MWV on the MDISK statement
 - all VSE systems must LINK..MW to that MDISK
- There can be NO DUPLICATE VOLIDS on the system
- In a DASD sharing complex all VSE systems must be properly shut down to release all locks (End-of-task in all partitions). Otherwise, the UNLOCK AR command may be used to free outstanding locks, or all systems must be shut down, and the first system to re-IPL should have a DLF...TYPE=F statement.



Important VM/CP Functions

VM CP Monitoring Commands

- Ipl <cuu>
- Ipl <cuu> CLEAR
- Display V<from>-<to>
- Display <addr>.<len>
- Display PwsG
- EXTernal
- STORE STATUS
- SYSTEM RESTART
- LOAD
- LOAD CLEAR
- Display range of virtual storage
- Display virtual storage for specified length
- Display PSW
- INTERRUPT function or IRPT Key
- Store Status
- System Restart



Important VM/CP Functions

VM CP TRACE Command

- TRace BR INTO <from>-<to>
- TRace I Range <from>-<to>
- TRace I/O <from>-<to>
- TRace STore INTO <addr>.<len>
- TRace SVC <code>
- TRace END ALL
- Trace branch instructions into specified target interval
- Trace instructions within specified interval
- Trace I/O devices in address range
- Trace storage alterations from address till length
- Trace SVCs with code
- reset all specified traces



Conclusion

Thank you for your attention ;-)