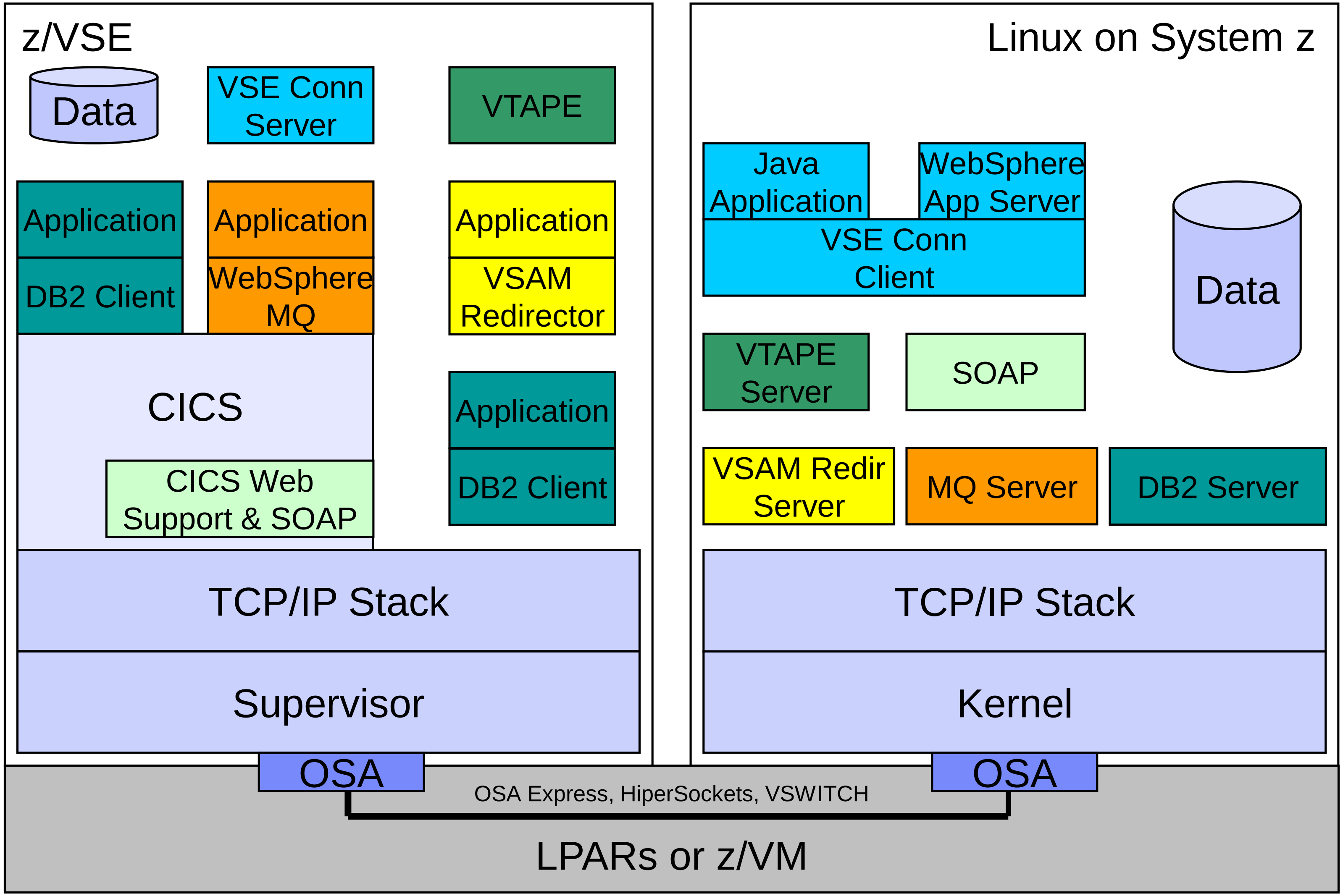
Implementing the z/VSE Fast Path to Linux on System z

Ingo Franzki, IBM

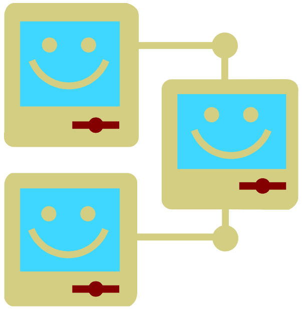
50 years of innovation

z/VSE Applications communicating with Applications on Linux



z/VSE Applications communicating with Applications on Linux

- **Communication is mostly based on TCP/IP**
  - Although z/VSE and Linux run on the same box



- **TCP/IP**
  - Allow reliable communication over a non-reliable network
  - Uses sequence numbers, acknowledges, checksums
    - To protect against packet loss, duplicate packets, packet sequence errors, damaged or incomplete packets, etc.

→ Time consuming processing

- **When z/VSE and Linux run side by side on the same box**
  - Why do we need all this expensive processing in this case?
  - There should be a more direct communication method !

→ z/VSE Fast Path to Linux on System z

(for short: Linux Fast Path or just LFP)

Fast Path to Linux on System z (LFP)

- **Allows selected TCP/IP applications to communicate with the TCP/IP stack on Linux without using a TCP/IP stack on z/VSE**
- **All socket requests are transparently forwarded to a Linux on System z system running in the same z/VM**

→ **Linux Fast Path in a z/VM environment**

- – Both z/VSE and Linux on System z run as z/VM Guests in the same z/VM-mode LPAR on IBM z10, z114 or z196 servers
- – Uses an IUCV connection between z/VSE and Linux

→ **Linux Fast Path in an LPAR environment**

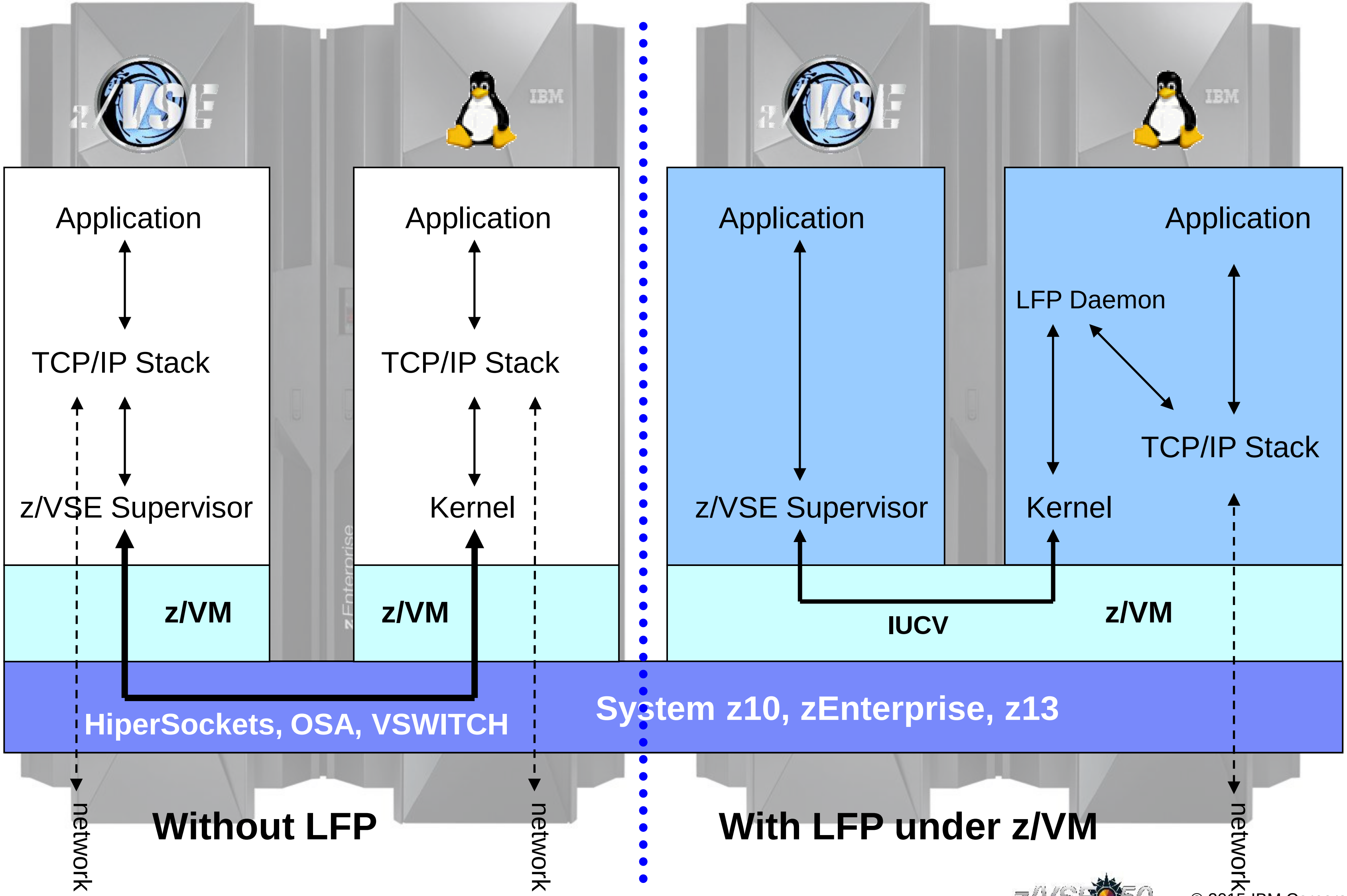
- – Both z/VSE and Linux on System z run in their own LPARs on a zEnterprise server
- – A HiperSockets connection is used between z/VSE and Linux on System z
- – LFP requires the HiperSockets Completion Queue function that is available with a zEnterprise server (z196, z114)

NEW!

- **The fast path to Linux on System z provides standard TCP/IP socket APIs for programs running on z/VSE**
  - – Other than the basic socket API, no other tools are provided
  - – Since z/VSE V5.1: LFP supports IPv6 NEW!
- **Possible performance increase due to:**
  - – Less overhead for TCP/IP processing on z/VSE (TCP, sequence numbers and acknowledging, checksums, resends, etc)
  - – More reliable communication method (IUCV) compared to HiperSockets, which is a network device, with all its packet drops, resends, etc.

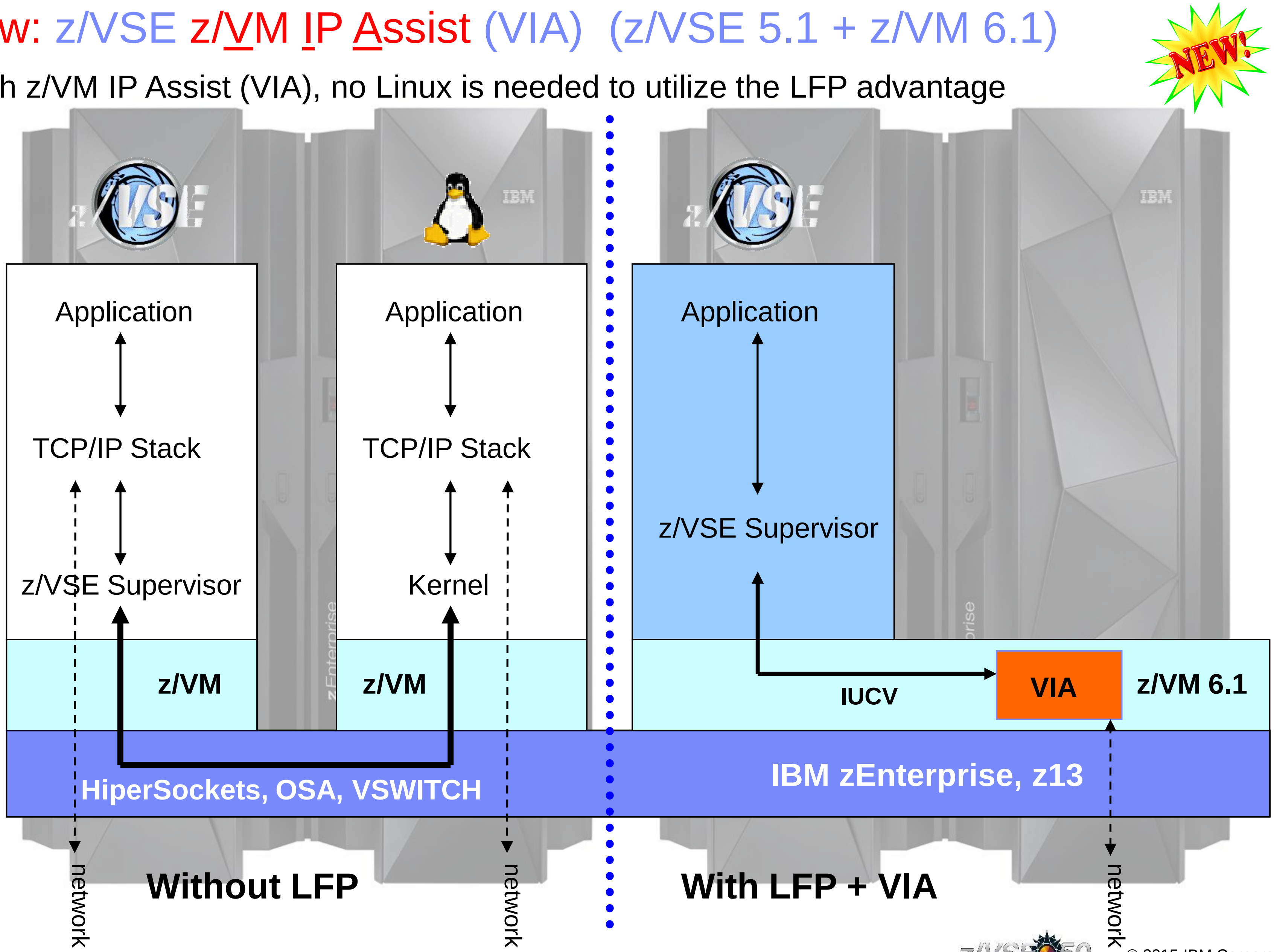
Linux Fast Path in a z/VM environment (z/VSE 4.3 or later)

Faster communication between z/VSE and Linux applications



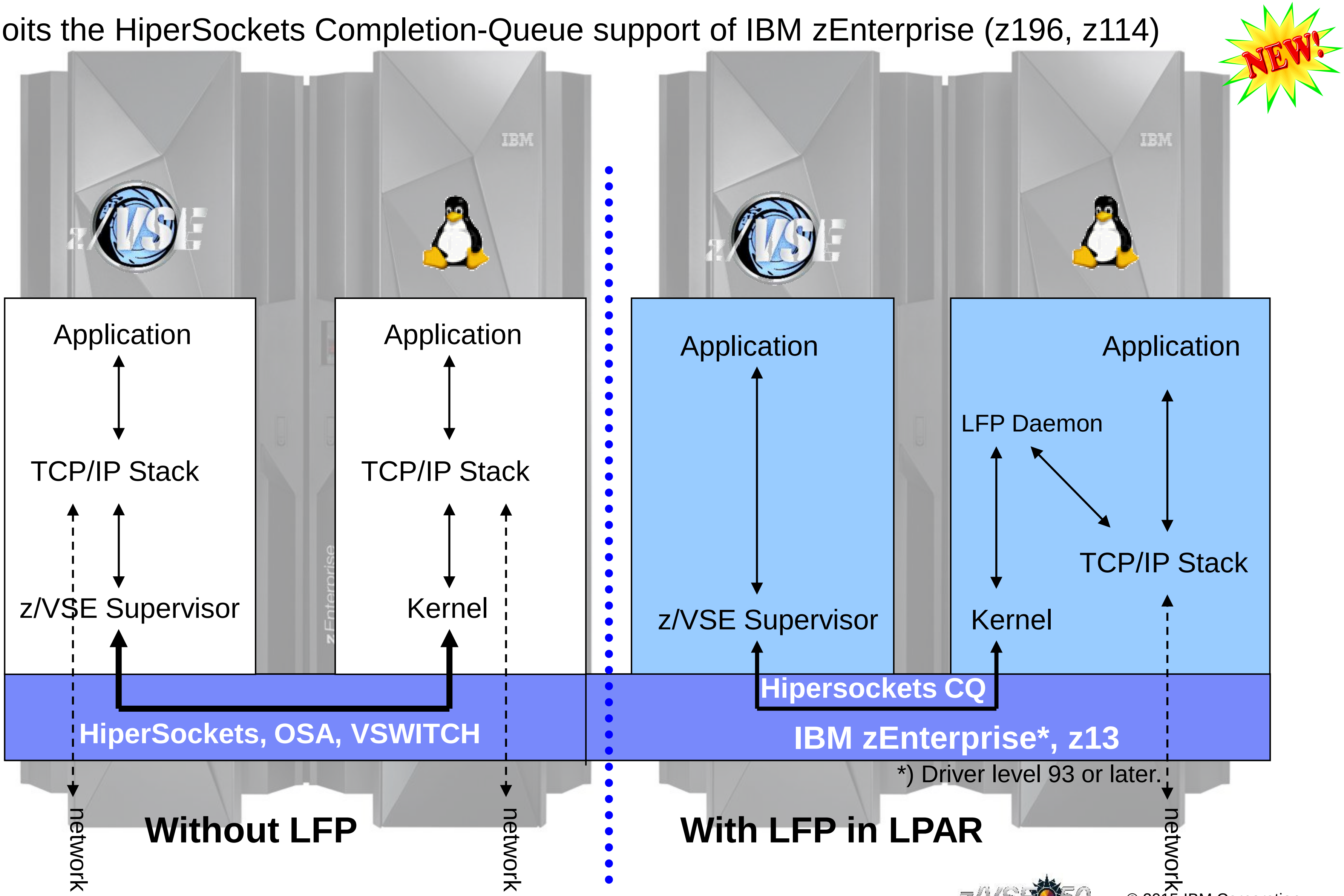
New: z/VSE z/VM IP Assist (VIA) (z/VSE 5.1 + z/VM 6.1)

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



New: Linux Fast Path in an LPAR environment (z/VSE 5.1 + PTFs)

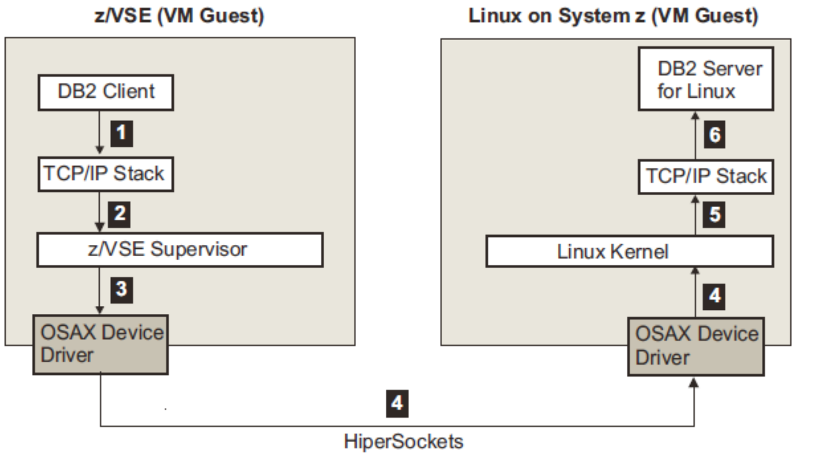
Exploits the HiperSockets Completion-Queue support of IBM zEnterprise (z196, z114)



*) Driver level 93 or later.

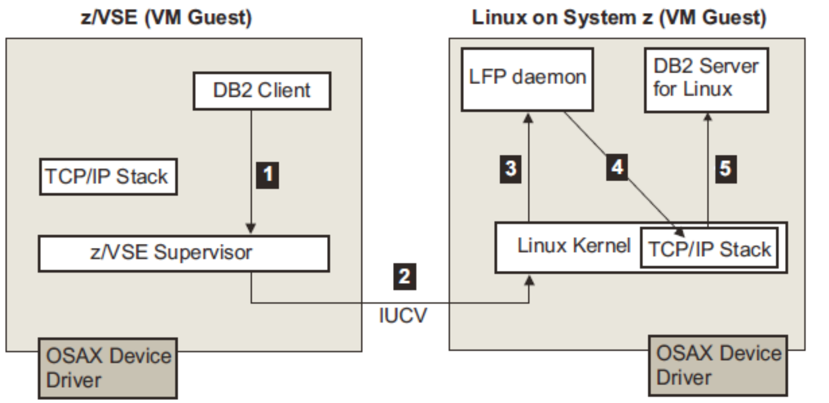
Communication flows when using Linux Fast Path

Using a TCP/IP stack (CSI/BSI):

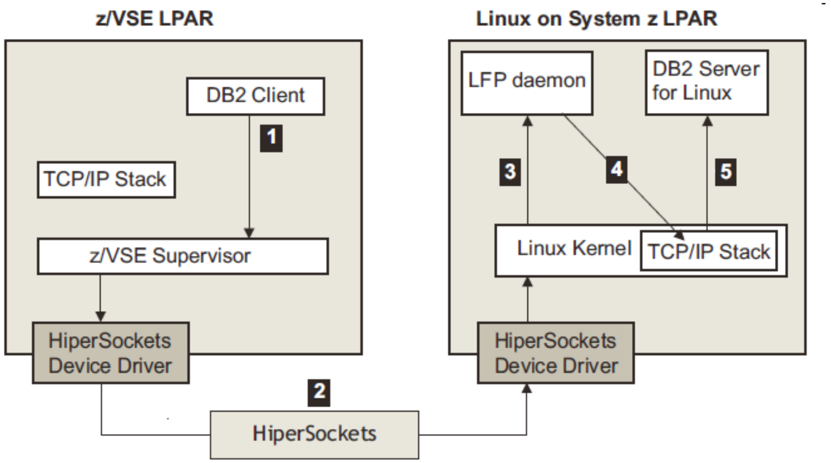


- **Less overhead for TCP/IP processing on z/VSE**
  - Building of IP and TCP packets
  - Sequence numbers and acknowledging
  - Checksums
  - Retransmission of lost packets
- **More reliable communication method compared to a traditional network device**
  - IUCV is a reliable communication method (z/VM)
  - HiperSockets Completion Queue support allows to build a reliable communication path (LPAR)

Using Linux Fast Path in a z/VM environment:



Using Linux Fast Path in an LPAR environment:



Prerequisites for using the Linux Fast Path

In a z/VM environment:

- **Any IBM System z server supported by z/VSE**
- **Any supported z/VM version/release**
- **The use of a z/VM-mode LPAR is recommended**
  - z/VM-Mode LPAR is only available on z10 or later and z/VM 5.4 or later

- **z/VSE 4.3 or later**

- **One of these Linux on System z operating systems:**
  - SUSE Linux Enterprise Server 10 Service Pack 3 together with security update kernel 2.6.16.60-0.57.1
  - SUSE Linux Enterprise Server 11 Service Pack 1
  - Red Hat Enterprise Linux 5 Update 5
  - Red Hat Enterprise Linux 6

- **z/VSE and Linux on System z are configured as z/VM guests within the same LPAR**

- **The IUCV (“Inter-User Communication Vehicle”) is configured and enabled in both z/VM guests (z/VSE and Linux on System z)**

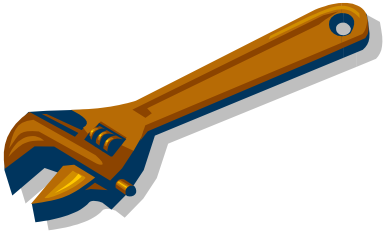
Prerequisites for using the Linux Fast Path

In an LPAR environment:

- **A zEnterprise server at driver level 93 or later or IBM z13**
  - LFP requires the HiperSockets Completion Queue function, which is only available with a zEnterprise server
- **z/VSE 5.1 + PTFs**
- **One of these Linux on System z operating systems:**
  - SUSE Linux Enterprise Server 11 Service Pack 2
  - Red Hat: IBM is working with its Linux distribution partners to include support in future Linux on System z distribution releases
- **z/VSE and Linux on System z both run in LPAR mode**
- **A HiperSockets Connection between z/VSE and Linux systems**

Preparing the system for Linux Fast Path in z/VM environment

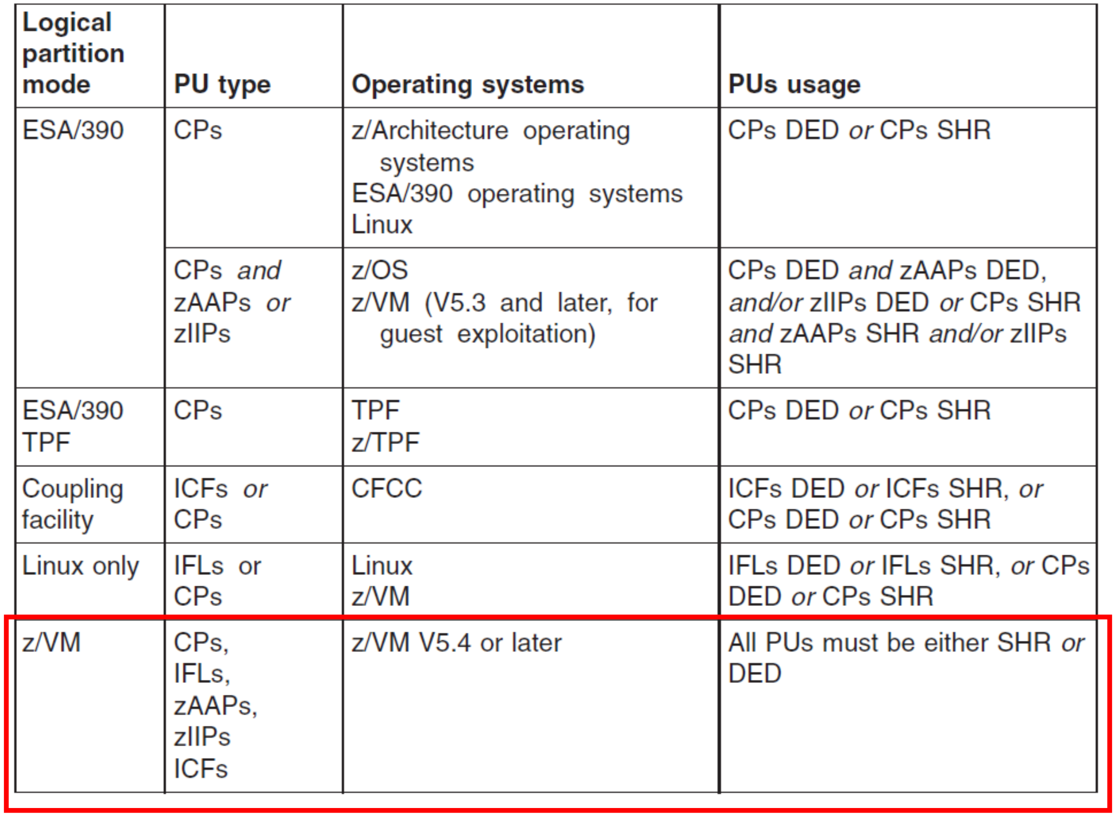
- **Preparing the LPAR**
  - For use with LFP in z/VM environment, the Linux on System z and z/VSE must run under the same z/VM system
  - The use of a z/VM Mode-LPAR is recommended
    - Allows you to mix CPs and IFL in one z/VM Installation
    - Linux runs on IFLs
    - z/VSE runs on CPs
  - Change the LPAR Mode to z/VM-Mode and add the IFLs to it



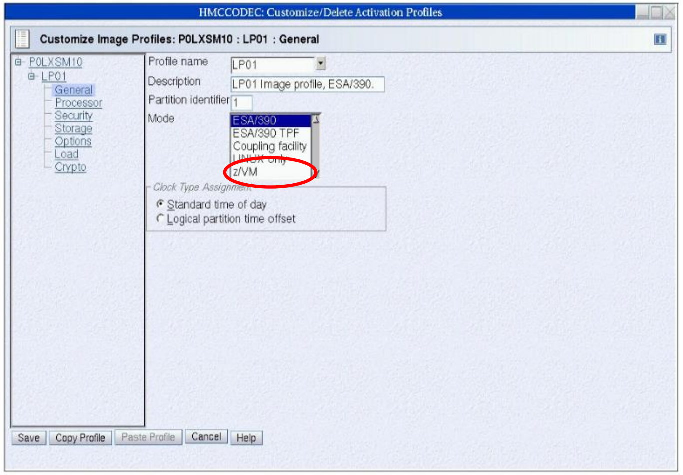
- **Preparing z/VM**
  - LFP uses IUCV as the underlying communication vehicle. Therefore the z/VSE and the Linux on System z guests on the z/VM system need to be configured for IUCV.
  - The following z/VM parameters for the guest systems are relevant:
    - IUCV ALLOW
    - IUCV ANY
    - IUCV MSGLIMIT
    - OPTION MAXCONN *maxno*
  - For details about the parameters check the z/VM documentation.

z/VM-Mode LPAR

LPAR Modes on z10 and later:



Logical partition mode	PU type	Operating systems	PUs usage
ESA/390	CPs	z/Architecture operating systems ESA/390 operating systems Linux	CPs DED *or* CPs SHR
	CPs *and* zAAPs *or* zIIPs	z/OS z/VM (V5.3 and later, for guest exploitation)	CPs DED *and* zAAPs DED, *and/or* zIIPs DED *or* CPs SHR *and* zAAPs SHR *and/or* zIIPs SHR
ESA/390 TPF	CPs	TPF z/TPF	CPs DED *or* CPs SHR
Coupling facility	ICFs *or* CPs	CFCC	ICFs DED *or* ICFs SHR, *or* CPs DED *or* CPs SHR
Linux only	IFLs or CPs	Linux z/VM	IFLs DED *or* IFLs SHR, *or* CPs DED *or* CPs SHR
z/VM	CPs, IFLs, zAAPs, zIIPs ICFs	z/VM V5.4 or later	All PUs must be either SHR *or* DED



HMCCODEC: Customize/Delete Activation Profiles

Customize Image Profiles: P0LXSM10 : LP01 : General

Profile name: LP01

Description: LP01 Image profile, ESA/390.

Partition identifier: 1

Mode: ESA/390

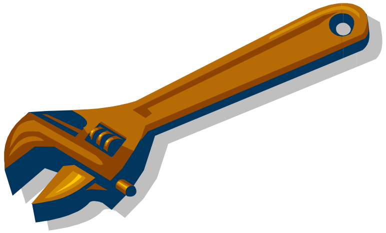
Clock Type Assignment

● Standard time of day
○ Logical partition time offset

Preparing to use Linux Fast Path

- **Preparing Linux on System z**
  - Download and install the LFP Daemon
    - Part of the “z/VSE Connector Workstation Code“ component 5686-CF8-38 / 51P
    - Member IJBLFPLX.W from PRD2.PROD or download from Internet
    - This ZIP file contains an RPM (RPM Package Manager) that can be used to install the LFPD
  - Configure one or multiple LFPD Instances
    - Textual configuration files in /etc/opt/ibm/vselfpd/confs-available and /etc/opt/ibm/vselfpd/confs-enabled
  - It is recommended to use separate (virtual) network adapters or at least separate IP addresses for each LFPD Instance (give each VSE its own IP address)
  - Start LFP daemon using lfpd-ctl or automatically at boot via init.d start script

- **Preparing z/VSE**
  - The LFP code is part of the z/VSE system, no installation step needed
  - Start and configure an LFP Instance
    - Textual configuration statements in LIBR member or SYSIPT of start job
    - LFP Instance operation via IJBLFPOP tool
  - LFP does not require a partition to run
  - Every LFP Instance is identified by a 2 digit number (System ID)
    - Same concept as used by TCP/IP stacks



Sample configuration on z/VSE

For LFP in z/VM Environment:

```
* $$ JOB JNM=LFPSTART,CLASS=0,DISP=L
// JOB LFPSTART
// EXEC IJBLFPOP,PARM=’START DD:SYSIPT LOGALL’
ID = 01
MTU = 8192
IucvMsgLimit = 1024
InitialBufferSpace = 512K
MaxBufferSpace = 4M
IucvSrcAppName = TESTV
IucvDestAppName = LINR02
IucvDestVMId = LINLFP
WindowSize = 65535
WindowThreshold = 25
/*
/&
* $$ EOJ
```

IJBLFPOP will read input from SYSIPT

IUCV Name of LFP on z/VSE

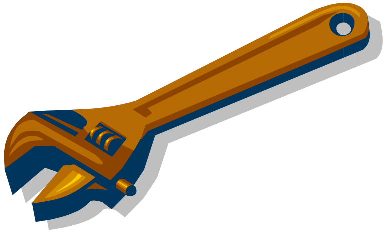
IUCV Name of LFPD on Linux

Guest name where Linux runs

z/VSE Skeletons for use with LFP

- **The following skeletons are available in ICCF library 59 for use with LFP:**

Skeleton	Description
SKLFPSTA	Start an LFP Instance
SKLFPSTO	Stop an LFP Instance
SKLFPLST	List all active LFP Instances
SKLFPINF	Query information about an active LFP Instance
SKLFPACT	Contains control statements to activate LFP you many need to include into the JCL of your applications



Operating an Linux Fast Path on z/VSE

- **List active LFP Instances**

  -
    ```
    // EXEC IJBLFPOP,PARM=’LIST’
    ```

  -
    ```
    LFPB025I ACTIVE LFP INSTANCES: 1
                     INSTANCE 01 HAS 3 ACTIVE TASKS
    LFPB026I END OF ACTIVE LFP INSTANCES LIST
    ```

- **Display information about an active instance**

  -
    ```
    // EXEC IJBLFPOP,PARM=’INFO <INSTID> [SHOWTASKS] [LOGALL]’
    ```

  -
    ```
    LFPB023I INFO ABOUT LFP INSTANCE ’01’:
      *** INSTANCE ***
        STATUS .......................... : UP
        WINDOW SIZE ..................... : 65,535
        ...
      *** DEVICE ***
        DEVICE STATUS ................... : ACTIVE
        PACKETS WAITING FOR MSG COMPLETE : 0
        MAXIMUM PACKETS USED ........... : 37
        ...
      *** TASKS ***
        ACTIVE TASK COUNT .............. : 3
       -- TASK #1 –
        TASK ID (PARTITION ID).......... : 2E (Z1)
        SOCKET COUNT ................... : 1
        L2 SOCKET LIST COUNT ........... : 1
        ...
    LFPB024I END OF INFO ABOUT LFP INSTANCE ’01’.
    ```

Sample configuration on Linux on System z

lfpd-LINR02.conf:

```
 # lfpd configuration file
IUCV_SRC_APPNAME = LINR02
# ensure that only TESTV from VSER05 can connect
PEER_IUCV_VMID = VSER05
PEER_IUCV_APPNAME = TESTV
IUCV_MSGLIMIT = 1024
MTU_SIZE = 8192
MAX_SOCKETS = 1024
INITIAL_IO_BUFS = 128
WINDOW_SIZE = 65535
WINDOW_THRESHOLD = 25
VSE_CODEPAGE = EBCDIC-US
VSE_HOSTID = 10.0.0.1
RESTRICT_TO_HOSTID = yes
LOG_INFO_MSG = no
```

IUCV Name of LFPD on Linux

Guest name where z/VSE runs

IUCV Name of LFP on z/VSE

This is the IP address VSE will appear under

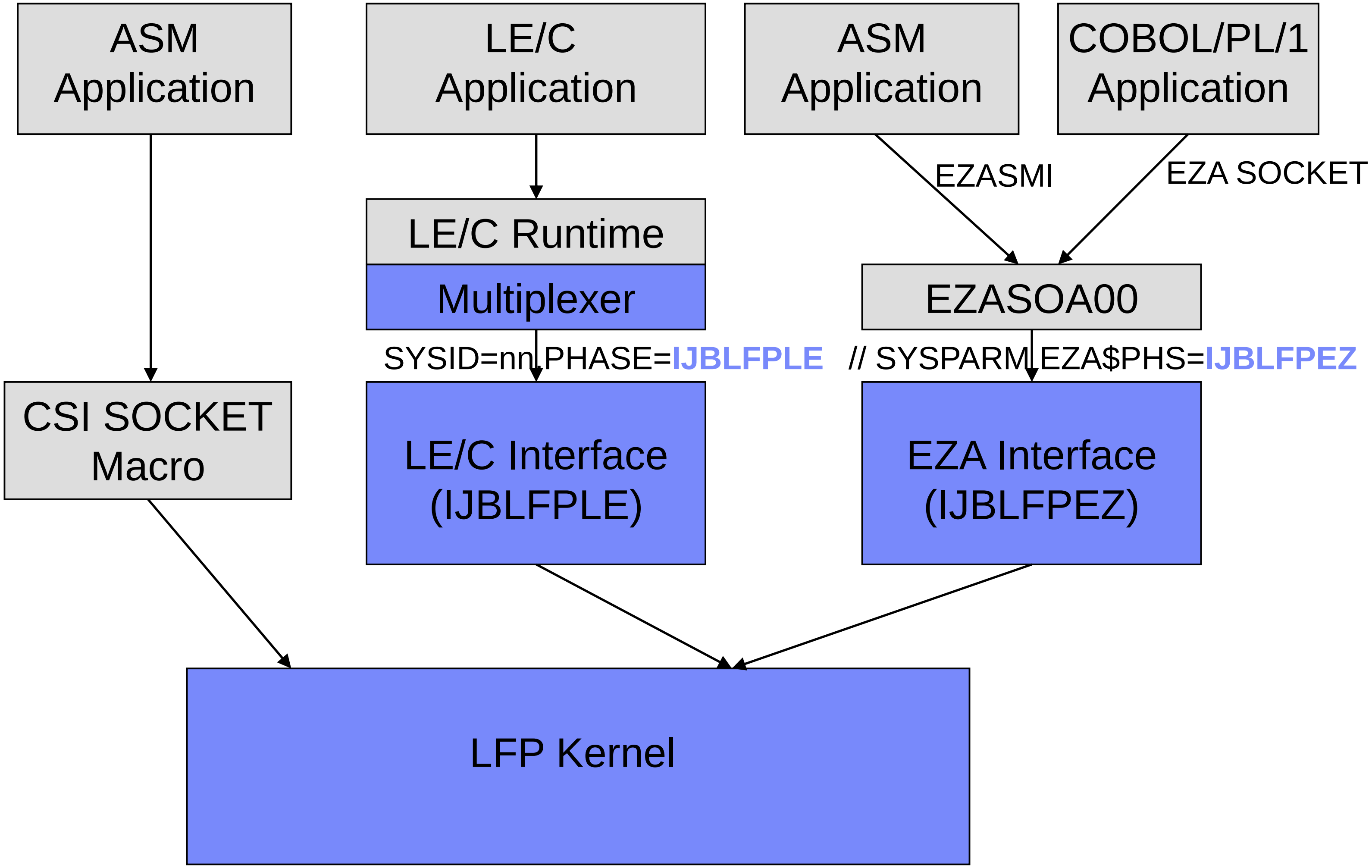
Note: The configuration file must be named “lfpd-XXX”, where XXX is the IUCV_SRC_APPNAME specified in the configuration file !
The XXX characters in the filename must be specified in uppercase !

Operating an Linux Fast Path on Linux on System z

- **Display LFP daemon status**
  - **`lfpd-admin <--iucv_appname|-i appname> <--status|-s>`**

```
Status:
-------
  z/VSE instance is connected.
  Peer VM ID ......... : VSER05
  Peer IUCV Appl. name : TESTV
  Applied host id .... : 10.0.0.1
  Applied host name .. : linlfp
  Allocated I/O buffers ....... : 128
  ...
  Number of active z/VSE tasks : 1
  Number of active sockets : 1
Trace Status:
-------------
  Running in daemon mode
  No trace is running
Configuration:
--------------
  LOCAL_IUCV_APPNAME = LINR02
  PEER_IUCV_VMID = VSER05
  PEER_IUCV_APPNAME = TESTV
  MAX_VSE_TASKS = 512
  MTU_SIZE = 8192
  MAX_SOCKETS = 1024
  INITIAL_IO_BUFS = 128
  WINDOW_SIZE = 65536
  WINDOW_THRESHOLD = 25% (16384 bytes)
  ...
-----------------
```

Socket API Support of Linux Fast Path



LE/C Socket API Multiplexer

- **Different Stacks use different Interface routines**
  - TCP/IP for VSE (CSI/IBM): $EDCTCPV
  - Linux Fast Path: IJBLFPLE
  - IPv6/VSE (BSI/IBM): BSTTTCP6

- **Avoid complicated setup using specific LIBDEFs for different stacks**
- **Interface phase is selected by System ID**
- **Use skeleton EDCTCPMC in ICCF library 62**

```
// EXEC ASMA90,SIZE=(ASMA90,64K),PARM='EXIT(LIBEXIT(EDECKXIT)),SIZE(MAXC
               -200K,ABOVE)'
EDCTCPMC CSECT
EDCTCPMC AMODE ANY
EDCTCPMC RMODE ANY
*
         EDCTCPME SYSID='00',PHASE='$EDCTCPV'
         EDCTCPME SYSID='01',PHASE='IJBLFPLE'
         EDCTCPME SYSID='02',PHASE='BSTTTCP6'
*
         END
/*
```

z/VSE Fast Path to Linux on System z (LFP)

- **Most existing applications run unchanged with Linux Fast Path**
  - Provided they use one of the supported Socket API (LE/C, EZA or ASM SOCKET)
    - And they do not use any CSI or BSI specific interface, features or functions
    - Since z/VSE V5.1: LFP supports IPv6 NEW!

- **IBM Applications supporting Linux Fast Path**
  - VSE Connector Server
  - CICS Web Support
  - VSE Web Services (SOAP) support (client and server)
  - CICS Listener
  - DB2/VSE Server and Client
  - WebSphere MQ Server and Client
  - VSAM Redirector
  - VSE VTAPE
  - VSE LDAP Support
  - VSE Script Client
  - POWER PNET
  - TCP/IP-TOOLS included in IPv6/VSE product (e.g. FTP Server/Client)

- **Customer applications should run unchanged:**
  - Provided they use one of the supported Socket API (LE/C, EZA or ASM SOCKET)

Performance measurements using Linux Fast Path

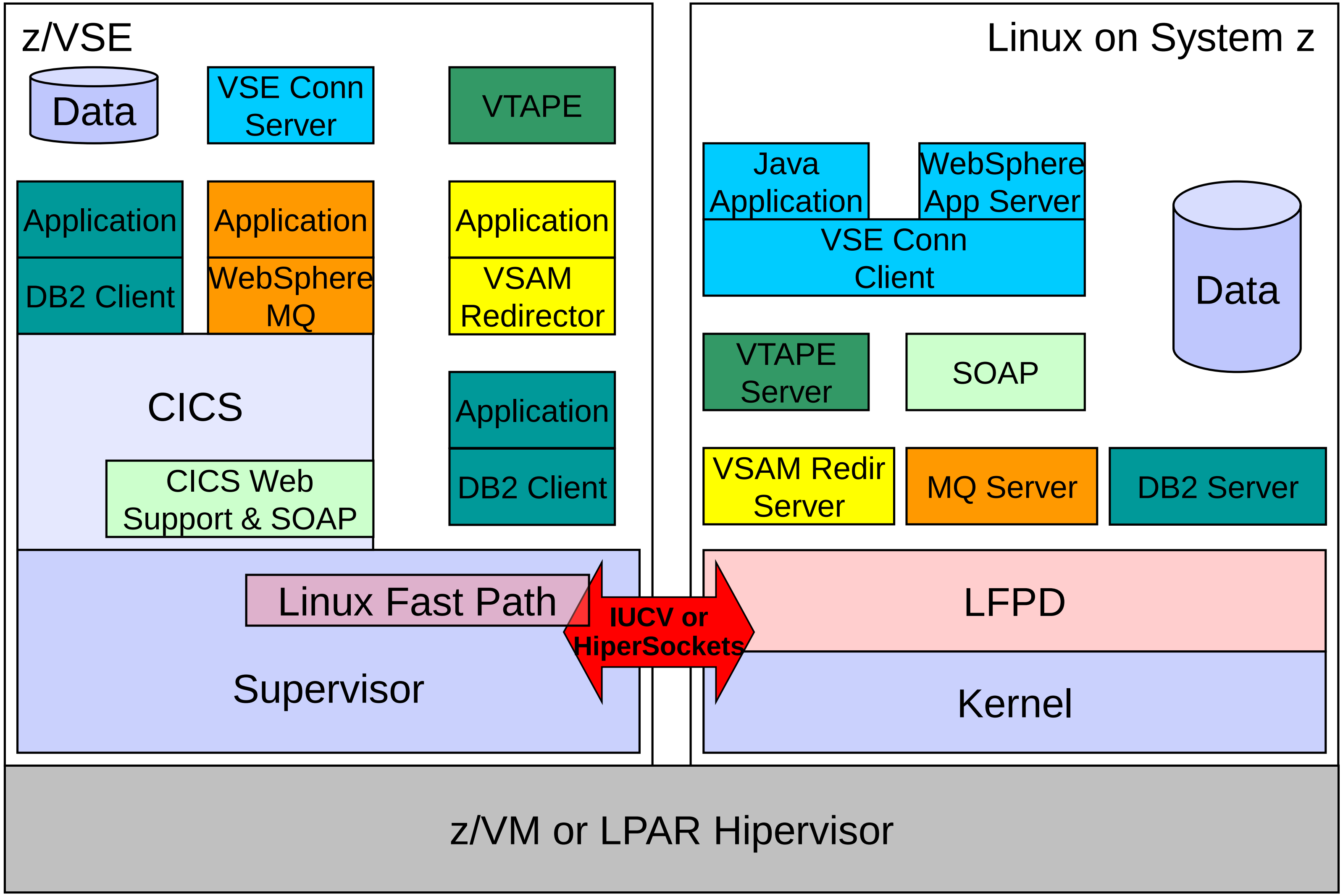
Comparison TCP/IP for VSE versus Linux Fast Path (z/VM Environment):

Workload	TCP/IP for VSE	Linux Fast Path (LFP)	Difference
FTP (BSI FTP server)			
▪VSE → Linux (1GB) (NULL file, no I/O)	19 MB/sec 29% CPU (5% App + 24% TCPIP)	72 MB/sec 20% CPU (App)	3.7 times faster 9% less CPU
▪Linux → VSE (1GB) (NULL file, no I/O)	21 MB/sec 55% CPU (11% App + 44% TCPIP)	70 MB/sec 20% CPU (App)	3.3 times faster 35% less CPU
Socket Application (running 3 times)			
▪VSE → Linux (100MB)	4.6 MB/sec (*3 = 13.8 MB/sec)	14.6 MB/sec (*3 = 43.8 MB/sec)	3.2 times faster
▪Linux → VSE (100MB)	9.7 MB/sec (*3 = 29.1 MB/sec)	16.2 MB/sec (*3 = 48.6 MB/sec)	1,7 times faster
	26% CPU (3*1% App + 23% TCP/IP)	9 % CPU (3*3% App)	17% less CPU

Environment: IBM System z10 EC (2097-722). TCP/IP connection via shared OSA adapter.

→Significant benefits in transfer rate as well as CPU usage
→Reduced Sub Capacity Cost

z/VSE Applications communicating with Applications on Linux



Questions ?

?

THANK YOU