



Linux on IBM System z and IBM FileNet P8 5.1
Setup, Performance, and Scalability

Table of Contents

About this publication	3
Introduction	4
Summary	5
System under test (SUT) overview	7
Hardware and software configuration	9
IBM zEnterprise 196 z/VM LPAR setup	9
IBM Storage subsystem setup	9
Network Setup	10
FileNet P8 Content Engine(CE) and Process Engine (PE) Server	10
FileNet P8 Database Server	11
FileNet P8 Application Engine (AE)	12
LDAP Security Server	13
FileNet P8 Enterprise Manager (FEM)	13
Client machine used as ECM workload driver	14
System under test (SUT) setup	14
z/VM setup	14
Virtual Networking	14
HyperPAV	15
Reorder Processing	15
Linux setup	15
Networking	16
Disk I/O	17
Other Linux operating system adaptations	21
FileNet P8 Content Engine (CE)	21
WAS V7 tuning for FileNet P8 Content Engine (CE)	22
FileNet P8 Process Engine (PE)	26
Tuning for the FileNet P8 Process Engine (PE)	26
DB2 CE/PE database server	27
Tuning the DB2 CE/PE database server	27
LDAP Security Server	29
Workload Description	30
Results	32
Workload scaling	32
CE/PE throughput and CPU load	33
CE average response times	34
PE average response times	35
CE/PE vertical scalability	36
CE/PE throughput and CPU load	36
CE average response times	38
PE average response times	40
Appendix	42
References	44
Notices	45

About this publication

This white paper contains information about the setup, performance and scalability of the IBM FileNet® P8 5.1 Enterprise Content Management (ECM) platform for Linux® on IBM System z® running under z/VM®. The FileNet P8 Content Engine and Process Engine tuning and an ECM performance study are described in this paper. Parts of the z/VM and Linux for System z setup are also discussed.

Authors

IBM System z Linux end-to-end Performance Team

Thomas Weber, Dr. Juergen Doelle

IBM FileNet P8 performance team

Darlene Jones, Jason Zhong, Dave Royer

Remarks

IBM Online Information Center web references pointing to specific product documentation chapters are usually very long, so the following notation is used in this paper. The web reference points to the top level website of the product documentation and a navigator below indicates the path to the specific topic for the product documentation.

For example:

<http://publib.boulder.ibm.com/infocenter/zvm/v6r2/index.jsp>

System Overview > z/VM V6R2 General Information > What is new or changed in z/VM 6.2 > Scalability > Disabling the guest page reordering process

Note: The web links referred in this paper are up-to-date as of June, 2013.

Introduction

IBM FileNet P8 provides a comprehensive, enterprise-class content lifecycle, document management, and workflow platform, for Enterprise Content Management (ECM) and Business Process Management (BPM). IBM FileNet P8 provides application APIs and tools to enable customers and business partners to create custom applications, as well as customizable out-of-the-box clients.

The P8 Content Engine (CE) server is designed to manage enterprise-wide documents and custom objects. It manages the classes, properties, storage, and metadata that form the foundation of an Enterprise Content Management system. It is installed on an application server with a database back-end on a separate guest. Document content can be stored either in the back-end database or, more commonly, in a filestore.

The P8 Process Engine (PE) server manages all aspects of business processes (also called workflows), such as process execution, process routing, rules management, process simulation and modelling, and workflow analysis. For the FileNet P8 5.1 release used in these tests, PE is a standalone Java™ application with a database back-end on a separate guest (same database as for CE in our environment). Note that starting with FileNet P8 version 5.2, CE and PE are now combined into the Content Platform Engine (CPE), running under control of the application server, and with a single combined CPE database.

This white paper presents the performance and scalability characteristics of the IBM FileNet P8 5.1 Content Engine (CE) and Process Engine (PE), in a virtualized environment running as Linux guests on z/VM on IBM System z. FileNet P8 is an ideal candidate for deployment in a virtualized environment, with all its flexibility for resource configurations and virtual network features where the communication between the components is implemented inside the local memory without hardware requirements. Performance tests driven using the CE and PE Java APIs demonstrate how FileNet P8 5.1 on Linux for System z can scale to effectively exploit an increasing number of Linux processors with good response times and throughput rates, with a workload that represents a realistic production transaction mix.

The performance results reported represent test data models and workloads run in an isolated laboratory on specific operating environments and system configurations. Actual performance in real customer environments with production workloads may vary significantly, and will depend on the unique circumstances of each customer's configuration and workload, and many factors including other applications running on the systems and configuration of the storage or network.

The following table shows the notational conventions used in this paper, in accordance to IEC 60027-2 Amendment 2.

Symbol	Full name	Derivation
KiB	kibibyte	2^{10} byte == 1024 byte
MiB	mebibyte	2^{20} byte == 1048576 byte
GiB	gibibyte	2^{30} byte == 1073741824 byte
KiB/s	kibibyte per second	2^{10} byte / second
MiB/s	mebibyte per second	2^{20} byte / second
GiB/s	gibibyte per second	2^{30} byte / second

Table 1. notational conventions

Summary

This paper describes in detail how IBM FileNet P8 5.1 was deployed and tuned in a virtualized environment on Linux on System z, on an IBM zEnterprise[®] 196 (z196) model 2817-M66 with 16 CPUs and 30 GiB central storage, connected to an IBM System Storage[®] DS8800 Model 951. It shows how the setup of the whole stack, including typical System z features such as HyperPAV and z/VM features as virtual networks, and various other Linux features and the FileNetP8 middleware are configured to create a high performing system.

The paper then describes the test workload, made of a realistic mix of the most common CE and PE transactions, and provides test results from scaling up this workload over a wide range of transaction rates, varying the number of system processors to evaluate how CPU sizing influences the performance.

The overall conclusion of the tests is that FileNet P8 5.1 shows excellent scalability on the Linux for System z system, with throughput increasing linearly until the CPUs of the CE/PE guest are nearly fully used. When scaling the CPUs, the throughput rate increase shows a nearly perfect scalability characteristics for the full workload bandwidth, which means when doubling the CPUs of the CE/PE guest a doubled workload can be reached. Overall we see a very nice symmetric multiprocessing (SMP) behaviour, which means that regardless how much virtual CPUs are configured for the CE/PE guest the used CPU capacity is the same. Having additional CPUs is not related with additional CPU cost due to increased management effort for the Linux kernel.

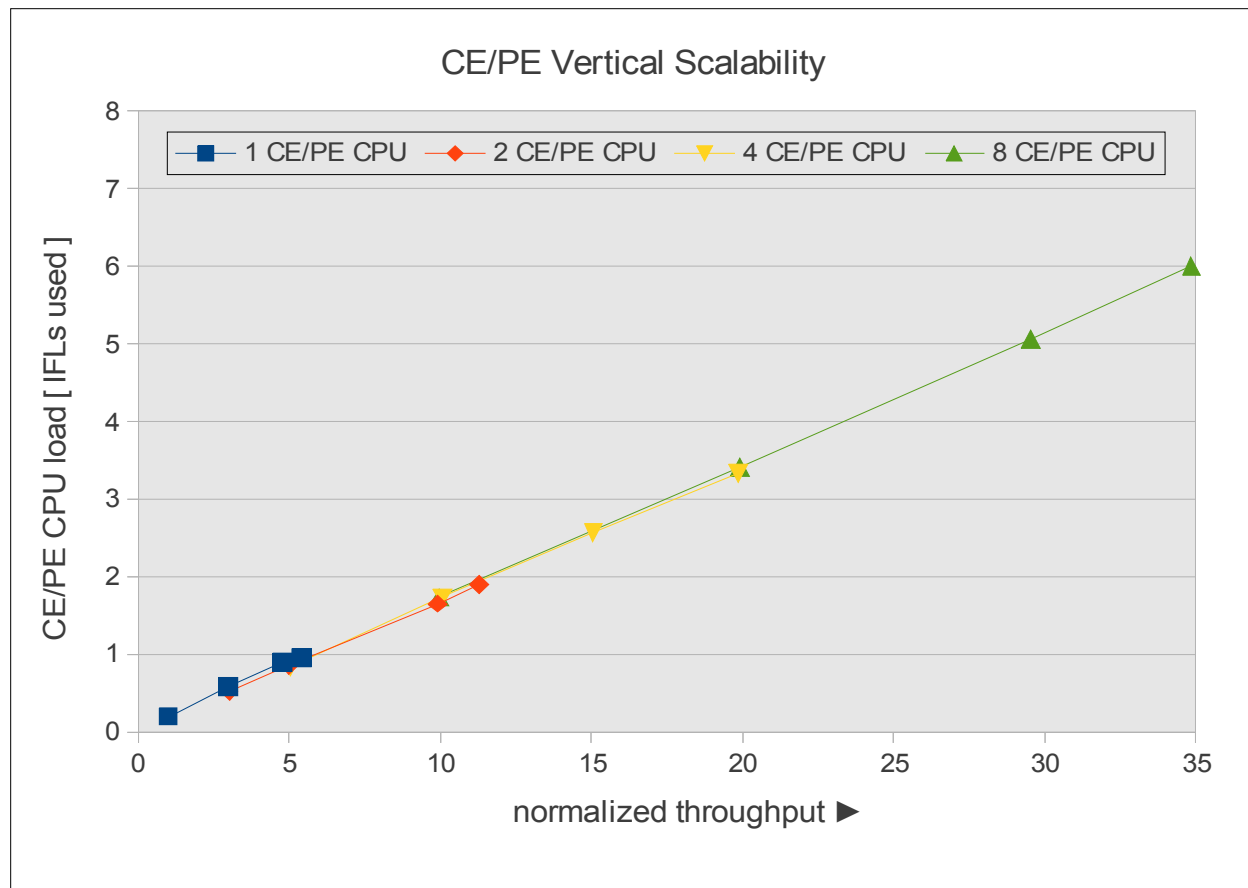


Figure 1. CE/PE vertical scalability – CE/PE CPU load with increasing transaction throughput

As illustrated in Figure 1, the normalized throughput achieved with 8 CE/PE CPUs was about 35x that of the lowest load tested with 1 CE/PE CPU, the base throughput normalized to “1”.

The paper provides a full set of performance and scalability test results. In general the average response times become much better as more CPUs are available on the CE/PE guest. The best response times are achieved with 8 CPUs. In the workload range where the CPUs are limiting the throughput (usually greater than 80% CPU load), the response times starts to increase. Adding further CPUs to such a CPU bound system takes the response times down again and allows further throughput increase. Having in mind that more virtual CPUs on the CE/PE guest for the same workload level are not related with additional CPU load, but with shorter response time, it indicates that the system benefits from a higher degree of parallelism.

The important message derived from these results is that for these types of workload the response times for a wide workload level range can be easily controlled by the amount of available CPUs on the CE/PE guest. The amount of CPUs on the database system has a minor role for the tested workload, with two CPs being sufficient for all scenarios, and for the lower workload levels even just one CPU was sufficient.

System under test (SUT) overview

This section introduces the Enterprise Content Management (ECM) solution and how it is implemented as system under test (SUT) for this project.

The SUT was set up with basic IBM FileNet P8 core components including a FileNet P8 Content Engine (CE) running as a Websphere® application, a FileNet P8 Process Engine (PE) and a FileNet P8 Application Engine (AE). These components together form an elementary ECM system also addressing Business Process Management (BPM) requirements.

The SUT implements a three-tier architecture model, enabling client components to interact with data resources and legacy applications.

Architecture model with logical tiers distributed across three independent systems:

- Client components running on local workstations (tier one)
- FileNet P8 application servers running on a remote server (tier two)
- Database and security servers running in the back-end on a remote server (tier three)

Tier one is responsible for the presentation and user interaction with second-tier processes. The client components enable the user to interact in a secure manner, by using FileNet P8 APIs to access the second tier servers. Tier-one client processes never access any third-tier services on the database server directly.

Tier one could be either an IBM Content Navigator client, Workplace™ XT client, or a custom client using the P8 Java APIs generating transactions against the FileNet CE/PE servers.

Tier two processes are also know as application logic layer. These processes manage the business logic of the application and provide access to third-tier services. It is the layer where most of the data processing occurs. CE and PE would represent the application logic layer for this setup. CE runs under the control of a Websphere Application server (WAS), while PE runs in a standalone Java JVM.

Tier three services are protected from direct access by any tier one components. Usually they are residing in a secure network (inside the System z box for example). Interaction is only possible through the second-tier processes. The DB2® database server and the LDAP security server are considered as tier three services in this case.

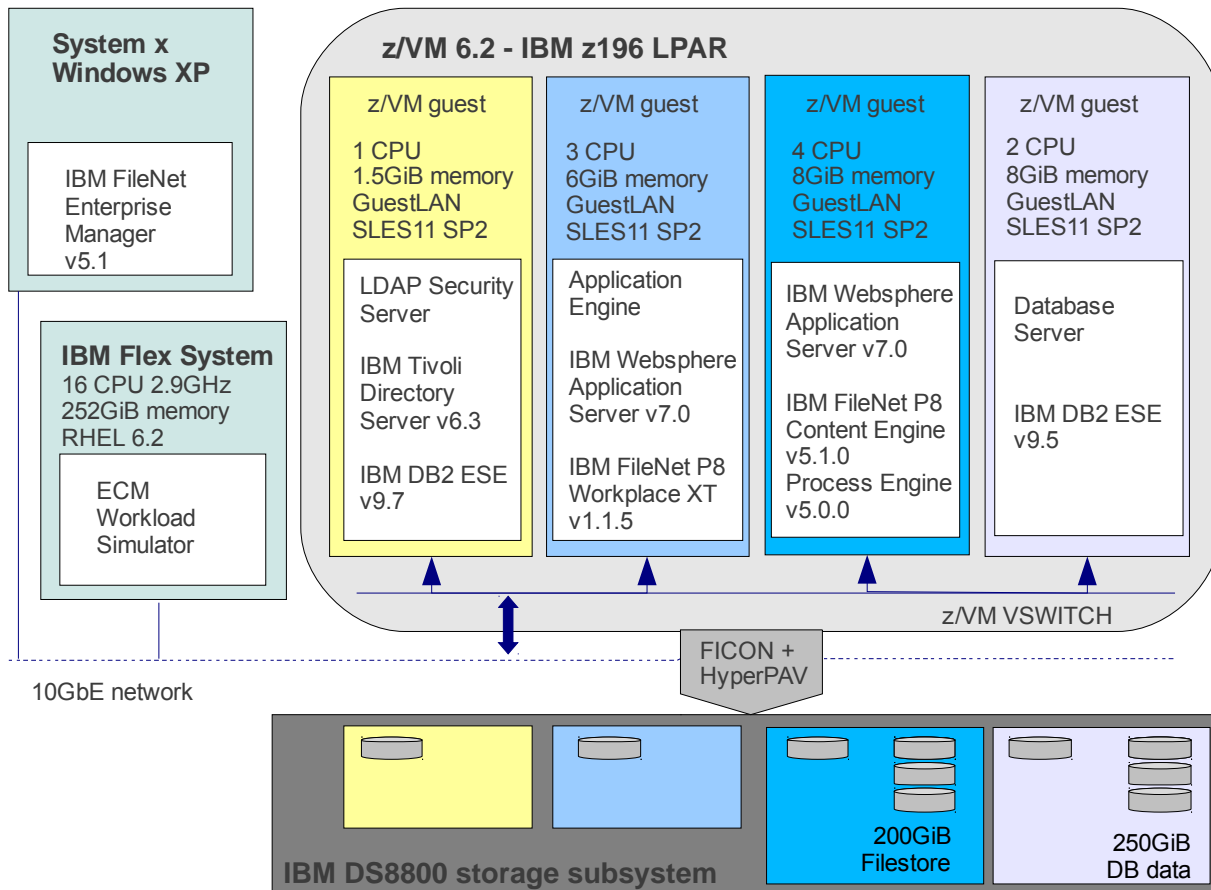


Figure 2. Overview of the system under test (SUT)

The system is set up in a virtualized environment using a z/VM hypervisor running in a IBM System z LPAR. The IBM FileNet P8 core components, the database server and LDAP security server were set up as four z/VM Linux guests in total. For inter-guest connectivity a z/VM guest LAN over a VSWITCH has been implemented. Client boxes outside the System z machine are connected via a 10GbE LAN. The DASD devices for the CE filesystem file store and the FileNet databases resided on an IBM System Storage DS8800 storage subsystem and also the DASD devices used for the z/VM hypervisor and the Linux guest operating systems. To provide a high I/O bandwidth storage pool striping was enabled for the storage subsystem. Further a couple of HyperPAV aliases are assigned to the CE/PE guest and the database server Linux guests, where higher disk I/O rates are expected for the SUT.

An IBM Flex System™ blade is used as the ECM workload driver. The workload simulator software uses the CE and PE Java APIs to emulate a custom application. The Flex blade is connected over the 10GbE network to the z196 server.

Hardware and software configuration

This chapter describes the software and hardware configurations for the System Under Test (SUT). The SUT represents an Enterprise Content Management (ECM) system consisting of IBM FileNet P8 core components. The system is set up as a three tier architecture including a

- FileNet P8 Content and Process Engine server
- DB2 database server
- Lightweight Directory Access Protocol (LDAP) security server
- IBM System x[®] client machine as ECM workload driver

The system was implemented using a single LPAR running a z/VM version 6.2 hypervisor on a System z Enterprise 196 (z196). The storage subsystem was a IBM DS8800 connected via FICON Express8.

IBM zEnterprise 196 z/VM LPAR setup

Single LPAR for z/VM version 6.2 on an IBM z196 model 2817-M66, equipped with:

- 16 Central Processors (CPs) supporting speeds up to 5.2 GHz
- 30 GiB Central Storage + 2 GiB Expanded Storage
- 1x 10GbE OSA-Express2 card
- 8x FICON Express8 S LX (short wave) features supporting ECKD[™] for DASD device access

IBM Storage subsystem setup

Storage subsystem with IBM System Storage DS8800 Model 951.

DASD disk devices are used for the z/VM Linux guest operating systems and ECM data.

- DASD Model 27, Model 54 and Model 128
- DASDs were selected from 2 ECKD storage pools (with storage pool striping enabled), alternatively selected from each internal server
- 40x HyperPAV alias devices available

FCP attached SCSI devices could also be used as disk attachment.

Network Setup

- 10 GbE LAN for all z/VM guests and for the System x client machine
- z/VM uses a Virtual Switch (VSWITCH) for guest-to-guest communication and a 10 GbE OSA-Express3 card attached to the VSWITCH for external connectivity
- no network encryption

FileNet P8 Content Engine(CE) and Process Engine (PE) Server

The following two tables list the z/VM guest resources and software versions for the CE/PE server:

z/VM guest setup
<ul style="list-style-type: none">• 1-8 virtual CPUs (varied for the CPU scalability tests)• 8 GiB memory• 10GbE LAN via VSWITCH• 1x DASD Model 27 for Linux operating system (20 GiB)• 2x DASD Model 128 combined as a Logical Volume(LV) (195 GiB) for the Content Engine(CE) file store• 20x HyperPAV aliases available• DASD devices and HyperPAV aliases are directly attached to the guest

Table 2.1. FileNet P8 CE/PE server z/VM guest resources

Software	Service level
<ul style="list-style-type: none">• Novell SUSE Linux Enterprise Server 11 for System z (64-bit)• IBM Websphere Application Server v7.0 for Linux on System z (64-bit)• IBM FileNet P8 Content Engine v5.1.0• IBM FileNet P8 Process Engine v5.0.0	<ul style="list-style-type: none">• Service Pack 2 (SP2)• Fix pack 23 (7.0.0.23)• Fix Pack 4 for CE (5.1.0.4)• Fix Pack 4 for PE (5.0.0.4)

Table 2.2. FileNet P8 server software

The FileNet P8 CE and PE engines run on the same z/VM guest and the CE/PE database backend server on a separate z/VM guest.

FileNet P8 Database Server

The following two tables list the z/VM guest resources and software versions for the FileNet P8 DB2 database server:

z/VM guest setup
<ul style="list-style-type: none">• 1-2 virtual CPUs (varied for the CPU scalability tests)• 8 GiB memory• 10 GbE LAN via VSWITCH• 1x DASD Model 27 for Linux operating system (20 GiB)• 4x DASD Model 54 combined as a Logical Volume(LV) (165 GiB) for FileNet P8 database data• 4x DASD Model 27 combined as a Logical Volume(LV) (82 GiB) for DB2 database log files• 2x DASD Model 54 combined as a Logical Volume(LV) (82 GiB) as DB2 backup space• 20x HyperPAV aliases available• DASD devices and HyperPAV aliases are directly attached to the guest

Table 3.1. FileNet P8 CE/PE Database server z/VM guest resources

Software	Service level
<ul style="list-style-type: none">• Novell SUSE Linux Enterprise Server 11 for System z (64-bit)• IBM DB2 Enterprise Server Edition V9.5	<ul style="list-style-type: none">• Service Pack 2 (SP2)• Fix Pack 10 (9.5.0.10)

Table 3.2. Database server software

The database server provides the backend databases for the CE and the PE engines.

FileNet P8 Application Engine (AE)

The following two tables list the z/VM guest resources and software versions for the FileNet P8 application engine:

z/VM guest setup
<ul style="list-style-type: none">• 3 virtual CPUs• 6 GB memory• 10 GbE LAN via VSWITCH• 1x DASD Model 27 for Linux operating system (20 GB) and application server installation• the DASD device is directly attached to the guest

Table 4.1. FileNet P8 Application Engine z/VM guest resources

Software	Service level
<ul style="list-style-type: none">• Novell SUSE Linux Enterprise Server 11 for System z (64-bit)• IBM Websphere Application Server v7.0 for Linux on System z (64-bit)• IBM FileNet P8 Workplace XT v1.1.5	<ul style="list-style-type: none">• Service Pack 2 (SP2)• Fix pack 23 (7.0.0.23)• 1.1.5.0

Table 4.2. Application Engine software

Workplace XT is an end-user web application that provides access to the document management capabilities of, and is tightly integrated with the business process management capabilities of, FileNet P8. Workplace XT is an optional FileNet P8 platform component (acting as Application Engine in our scenario). It provides access to the process and content functionality of FileNet P8 CE/PE. Workplace XT can be installed in addition to or in place of an Application Engine (AE). Workplace XT protects user credentials passed between Workplace XT and Content Engine and, if configured, provides SSL security.

Note: the application engine was not used during the measurements, but was used for setup and verification purposes.

LDAP Security Server

The following two tables list the z/VM guest resources and software versions for the LDAP Security server:

z/VM guest setup
<ul style="list-style-type: none">• 1 virtual CPU• 1.5 GiB memory• 10 GbE LAN via VSWITCH• 1x DASD Model 27 (20GiB) for Linux operating system and LDAP security server installation• The DASD device is directly attached to the guest

Table 5.1. LDAP Security server z/VM guest resources

Software	Service level
<ul style="list-style-type: none">• Novell SUSE Linux Enterprise Server 11 for System z (64-bit)• IBM Tivoli[®] Directory Server v6.3<ul style="list-style-type: none">– ships IBM DB2 ESE v9.7	<ul style="list-style-type: none">• Service Pack 2 (SP2)• 6.3.0.0

Table 5.2. LDAP Security server software

The LDAP server manages the authentication of the test users.

FileNet P8 Enterprise Manager (FEM)

The following table lists the software versions for the System x client machine with the FileNet P8 Enterprise Manager installation:

Software	Service level
<ul style="list-style-type: none">• Windows XP Professional• IBM FileNet P8 Enterprise Manager v5.1.0	<ul style="list-style-type: none">• 5.1.2600• 5.1.0

Table 6. Client software

The FileNet Enterprise Manager (FEM) is one of the primary tools for administering the Content Engine (CE). It provides access to most of the CE administrative functionality and was required to administration tasks like creating a P8 domain, a CE object store, a PE region ID and a PE connection point, however FEM was not required during the measurements.

Note: FEM cannot be installed on a UNIX[®] machine for FileNet P8 CE, Version 5.1 and needs to be installed on a supported Microsoft[®] Windows[®] system.

Client machine used as ECM workload driver

The following two tables list the hardware and software versions for the ECM workload driver machine:

Client hardware	Setup
IBM Flex System	<ul style="list-style-type: none">• 16 CPUs (2.9 GHz)• 250 GiB memory• 10 GbE network card for application server connectivity

Table 7.1. Client hardware

Software	Service level
<ul style="list-style-type: none">• Red Hat Enterprise Linux Server Release 6• ECM Workload Simulator (Java API)	<ul style="list-style-type: none">• Service Level 2 (Santiago)

Table 7.2. Client software

This system acts as the workload driver in the SUT. It generates the load with a custom application that uses the FileNet CE and PE Java APIs to generate a customer-like transaction mix.

System under test (SUT) setup

Setup modifications applied to the individual components of the SUT are described in this chapter. Changes were done for z/VM, Linux for System z and some of the IBM FileNet P8 core components.

z/VM setup

All z/VM guest resources defined for the various servers are outlined in detail in the 'Hardware and software configuration' chapter. Besides the z/VM resource allocations, the following tuning and additional configuration was applied.

Virtual Networking

The z/VM guest's use a Virtual Switch (VSWITCH) configured LAN for guest-to-guest communication. A 10 GbE OSA-Express3 is connected to the VSWITCH and the guests attached to the VSWITCH reside in the same LAN as the 10GbE OSA Express card is attached to. This allows the configuration of guest IP addresses from the same network that are used by the OSA Express card. As a virtual networking solution, the VSWITCH provides good performance and is the recommended method for internal and external z/VM network connectivity.

See also Linux on System z network connection types presentation:

http://www.ibm.com/developerworks/linux/linux390/perf/tuning_networking.html#nct

For more information about z/VM virtual networking and the VSWITCH you can start here:

<http://www.vm.ibm.com/virtualnetwork/>

HyperPAV

The I/O throughput for DASDs can be improved by using Parallel Access Volumes (PAV) or HyperPAV (see also Linux setup chapter).

The FileNet P8 CE/PE server and the DB2 database server are candidates for doing a lot of disk I/O. Both have got 20 HyperPAV aliases directly attached to the guest, which are available for use with the Linux operating system then.

Reorder Processing

z/VM triggers from time to time a processing to reorder the pages for all its virtual machines. While the page reorder processing runs, the virtual machine is stopped. This behaviour is not wanted when running a benchmark application over a longer period, because it affects the results of the benchmark application.

It is possible to disable the page reorder processing with the SET REORDER command for the entire z/VM operating system. In some situations it makes sense to disable page reorder processing aside from benchmarking.

Check out the IBM z/VM documentation about “Disabling the guest page reordering process”:

<http://publib.boulder.ibm.com/infocenter/zvm/v6r2/index.jsp>

System Overview > z/VM V6R2 General Information > What is new or changed in z/VM 6.2 > Scalability > Disabling the guest page reordering process

For details about the Reorder Processing, see:

<http://www.vm.ibm.com/perf/tips/reorder.html>

Linux setup

The guest resource definitions for the Linux servers in the ECM system are outlined in detail in the 'Hardware and software configuration' chapter.

Besides that the following additional operating system tuning and setup changes were applied.

Networking

The FileNet P8 servers under z/VM communicate over a virtual guest-to-guest network using a z/VM Virtual Switch (VSWITCH). The virtual network device appears as 'GuestLAN QDIO' type in Linux when listing the devices with the *lsqeth* command.

Sample command: *lsqeth* for the virtual network device

```
# lsqeth eth1
Device name           : eth1
-----
card_type             : GuestLAN QDIO
cdev0                 : 0.0.1000
cdev1                 : 0.0.1001
cdev2                 : 0.0.1002
chpid                 : 02
online                : 1
portname              : osaport
portno                : 0
route4                : no
route6                : no
checksumming          : sw checksumming
state                 : UP (LAN ONLINE)
priority_queueing     : always queue 2
fake_broadcast        : 0
buffer_count          : 128
layer2                : 0
large_send            : no
isolation              : none
sniffer               : 0
```

The buffer count for the network devices was increased for all FileNet servers:

- Increased the number of buffers for inbound traffic to 128 (default is 64) at `/etc/udev/rules.d/51-qeth-0.0.1000.rules`

where '1000' is first cdev of the virtual network device in this case

- The following line was added to the udev rule definition to enhance the buffer count:

```
ACTION=="add", SUBSYSTEM=="ccwgroup", KERNEL=="0.0.1000", ATTR{buffer_count}="128"
```

See also Linux on System z – Tuning hints and tips

http://www.ibm.com/developerworks/linux/linux390/perf/tuning_networking.html

Disk I/O

Using HyperPAV

The I/O throughput for an ECKD DASD disk device can be improved by using Parallel Access Volumes (PAV) or HyperPAV. This feature is important for DASDs used for the FileNet P8 CE file store and for the FileNet P8 CE/PE databases where a lot of disk I/O occurs.

The Linux DASD device driver can use this IBM System Storage feature to perform multiple concurrent data transfer operations to or from the same DASD device instead of single data transfers. To use HyperPAV, there must be base and alias devices available, which require System z Input/Output Configurations Data Set (IOCDS) definitions. For HyperPAV on an IBM System Storage subsystem, the alias devices are not exclusively referenced to a certain base device, but they are eligible for all base devices in the same logical control unit (LCU). Linux handles HyperPAV alias devices in the same way as a normal DASD base devices by using the *chccwdev* command or defining the appropriate udev rules for them. When listing the DASD devices with the *lsdasd* command, HyperPAV aliases can be identified per the 'alias' status tag. The usage of the HyperPAV aliases is completely handled by the Linux kernel and transparent to the users.

Sample command: `lsdasd` showing 11 DASD devices and 20 HyperPAV aliases for the database server

#	lsdasd	Bus-ID	Status	Name	Device	Type	BlkSz	Size	Blocks
		0.0.6ebc	alias			ECKD			
		0.0.6ebd	alias			ECKD			
		0.0.6ebe	alias			ECKD			
		0.0.6ebf	alias			ECKD			
		0.0.6ec0	alias			ECKD			
		0.0.6ec1	alias			ECKD			
		0.0.6ec2	alias			ECKD			
		0.0.6ec3	alias			ECKD			
		0.0.6ec4	alias			ECKD			
		0.0.6ec5	alias			ECKD			
		0.0.6fbc	alias			ECKD			
		0.0.6fbd	alias			ECKD			
		0.0.6fbe	alias			ECKD			
		0.0.6fbf	alias			ECKD			
		0.0.6fc0	alias			ECKD			
		0.0.6fc1	alias			ECKD			
		0.0.6fc2	alias			ECKD			
		0.0.6fc3	alias			ECKD			
		0.0.6fc4	alias			ECKD			
		0.0.6fc5	alias			ECKD			
		0.0.7215	active	dasda	94:0	ECKD	4096	21129MB	5409180
		0.0.680d	active	dasdb	94:4	ECKD	4096	42259MB	10818360
		0.0.690c	active	dasdc	94:8	ECKD	4096	42259MB	10818360

0.0.6e0c	active	dasdd	94:12	ECKD	4096	42259MB	10818360
0.0.6e0d	active	dasde	94:16	ECKD	4096	42259MB	10818360
0.0.6e0f	active	dasdf	94:20	ECKD	4096	21129MB	5409180
0.0.6e10	active	dasdg	94:24	ECKD	4096	21129MB	5409180
0.0.6f0c	active	dasdh	94:28	ECKD	4096	42259MB	10818360
0.0.6f0d	active	dasdi	94:32	ECKD	4096	42259MB	10818360
0.0.6f0f	active	dasdj	94:36	ECKD	4096	21129MB	5409180
0.0.6f10	active	dasdk	94:40	ECKD	4096	21129MB	5409180

For details about IOCDS configurations, see:

Input/Output Configuration Program User's Guide, SB10-7037-10

<http://www.ibm.com/servers/resourcelink> (registration required)

For more information about PAV and HyperPAV, see:

How to Improve Performance with PAV, SC33-8414

http://www.ibm.com/developerworks/linux/linux390/development_documentation.html

A detailed study addressing HyperPAV and databases can be found in this IBM white paper

“Oracle Database on Linux on System z - Disk I/O Connectivity Study”

http://www.ibm.com/developerworks/linux/linux390/perf/tuning_database.html#con

Using the Logical Volume Manager

The Linux Logical Volume Manager (LVM) was used to create Logical Volumes (LV) using a couple of physical DASD devices. The LVs were defined with striping enabled, so that any I/O operations can be parallelized across the physical DASD devices within the LV. This allows a higher performance for reading and writing sequential files, but also benefits for random disk I/O. LVs are used for the FileNet P8 CE/PE server and the database server for their large data. The underlying Volume Groups (VG) were set up with full DASD devices.

Sample command: pvscan showing the DASDs assigned to database server VGs

```
# pvscan
PV /dev/dasdf1 VG DB_log_vg lvm2 [20.63 GiB / 0 free]
PV /dev/dasdj1 VG DB_log_vg lvm2 [20.63 GiB / 0 free]
PV /dev/dasdg1 VG DB_log_vg lvm2 [20.63 GiB / 0 free]
PV /dev/dasdk1 VG DB_log_vg lvm2 [20.63 GiB / 0 free]
PV /dev/dasdd1 VG ECM_data_vg lvm2 [41.27 GiB / 0 free]
PV /dev/dasde1 VG ECM_data_vg lvm2 [41.27 GiB / 0 free]
PV /dev/dasdh1 VG ECM_data_vg lvm2 [41.27 GiB / 0 free]
PV /dev/dasdi1 VG ECM_data_vg lvm2 [41.27 GiB / 0 free]
PV /dev/dasdb1 VG ECM_backup_vg lvm2 [41.27 GiB / 0 free]
PV /dev/dasdc1 VG ECM_backup_vg lvm2 [41.27 GiB / 0 free]
Total: 10 [330.11 GiB] / in use: 10 [330.11 GiB] / in no VG: 0 [0]
```

Sample command: `lvdisplay` showing the extent mapping of the database data LV

```
# lvdisplay -m /dev/ECM_data_vg/ECM_data_lv
--- Logical volume ---
LV Name                /dev/ECM_data_vg/ECM_data_lv
VG Name                ECM_data_vg
LV UUID                dtEdy3-TZGY-5dcY-wMv1-Scx1-XWHx-4fzHGv
LV Write Access        read/write
LV Status              available
# open                 1
LV Size                165.06 GiB
Current LE             42256
Segments              1
Allocation             inherit
Read ahead sectors    auto
- currently set to    1024
Block device          253:1

--- Segments ---
Logical extent 0 to 42255:
  Type                striped
  Stripes              4
  Stripe size          64.00 KiB
  Stripe 0:
    Physical volume    /dev/dasdd1
    Physical extents   0 to 10563
  Stripe 1:
    Physical volume    /dev/dasde1
    Physical extents   0 to 10563
  Stripe 2:
    Physical volume    /dev/dasdh1
    Physical extents   0 to 10563
  Stripe 3:
    Physical volume    /dev/dasdi1
    Physical extents   0 to 10563
```

The above LVM command outputs show the setup for the database server. The database server has three VGs each with one LV in total for database data files, database log files and one for backup purposes. It is common to put the database log files and the database data files on separate Logical Volumes to avoid that they interfere together. The VG to LV one-to-one relation implicates that each LV has its own DASD devices. The DASDs were alternately selected from two ECKD storage pools, so that the LV can benefit from both storage subsystem internal server caches.

The `lvdisplay` command for the database data LV shows four physical extents (full DASDs in this case). From performance point of view, it is recommended to define the number of stripes equal to the number of extents in the LV. Hence four LV stripes are used in this example. Further a stripe size of 64 KiB has been chosen.

Sample command: `pvs` listing the DASDs assigned to the file store VG for the FileNet P8

Content Engine

```
# pvscan
PV /dev/dasdb1   VG CE_filestor_vg   lvm2 [97.82 GiB / 0   free]
PV /dev/dasdc1   VG CE_filestor_vg   lvm2 [97.82 GiB / 0   free]
Total: 2 [195.63 GiB] / in use: 2 [195.63 GiB] / in no VG: 0 [0   ]
```

Sample command: `lvdisplay` showing the extent mapping of the DB data LV

```
# lvdisplay -m
--- Logical volume ---
LV Name                /dev/CE_filestor_vg/CE_filestor_lv
VG Name                CE_filestor_vg
LV UUID                8xpcgE-Y14U-0hpe-YWPa-Q2wv-Gk6X-FYM9Ji
LV Write Access        read/write
LV Status              available
# open                 1
LV Size                195.63 GiB
Current LE             50082
Segments              1
Allocation             inherit
Read ahead sectors    auto
- currently set to    1024
Block device          253:0

--- Segments ---
Logical extent 0 to 50081:
  Type                striped
  Stripes             2
  Stripe size         64.00 KiB
  Stripe 0:
    Physical volume   /dev/dasdb1
    Physical extents  0 to 25040
  Stripe 1:
    Physical volume   /dev/dasdc1
    Physical extents  0 to 25040
```

The above LVM command outputs show the LV setup for the FileNet P8 CE/PE server. One LV was created for the FileNet P8 CE file store. The LV stripe size is set to two using stripe size of 64 KiB

For more information for Linux on System z disk I/O tuning, see:

http://www.ibm.com/developerworks/linux/linux390/perf/tuning_diskio.html#dpo

Other Linux operating system adaptations

Websphere Application Server (WAS) V7 installation

The FileNet P8 Application Engine (AE) and Content Engine (CE) are applications running under the control of a Websphere Application Server (WAS). The WAS installation process and console requires a supported web browser to be installed (for example Mozilla Firefox).

Typically not all X11 packages are installed in a Linux for System z installation. One possible way to export Graphical User Interfaces (GUI) for installation process steps is to display them on your local workstation. One possibility is to use the Virtual Network Protocol (VNC) for that purpose even it is functionally not required. This requires the `xorg-x11-xvnc` package installed on the server. You can use the `ssh` command to log into servers with X11 forwarding enabled then (`ssh -X user@server`).

WAS V7 requires an increased user limit (`ulimit`) for open files. The new limit was set for the FileNet P8 CE/PE server and application engine (AE) server where a WAS application runs. The maximum limit for the number of open files or open file descriptors was set to 8192:

- by adding the `ulimit` command to `/etc/profile.local` to make the change a default for newly opened shells

```
ulimit -n 8192
```

For the 64-bit SUSE Linux Enterprise Server Version 11 operating system, both the `xorg-x11-libs` and the `xorg-x11-libs-32bit` package must be present before installing and using WebSphere Application Server.

FileNet P8 Content Engine (CE)

After deploying the FileNet P8 CE application you can verify that the CE deployment was successful by accessing the FileNet P8 CE health page and ping page.

Browse the FileNet P8 CE health page:

```
http://<server>:<port>/P8CE/Health
```

Browse the FileNet P8 CE startup context (also known as CE ping page):

```
http://<server>:<port>/FileNet/Engine
```

Note: In case of a WebSphere Application Server the default port is usually 9080.

For more information, refer the FileNet P8 Version 5.1 Information Center:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

Installing or upgrading IBM FileNet P8 Platform > Installing a distributed IBM FileNet P8 Platform system > Installing and configuring Content Engine > Verifying the Content Engine deployment

WAS V7 tuning for FileNet P8 Content Engine (CE)

WAS Java Virtual Machine settings

The CE Java heap size was enlarged for the ECM benchmark application tests to 2 GiB. The initial/maximum heap size for CE was set to **2048 MiB** and the following generic JVM arguments have been added:

Initial heap size	<input type="text" value="2048"/>	MB
Maximum heap size	<input type="text" value="2048"/>	MB
<input type="checkbox"/> Run HProf		
HProf Arguments	<input type="text"/>	
<input type="checkbox"/> Debug Mode		
Debug arguments	<input type="text" value="-agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=7777"/>	
Generic JVM arguments	<input type="text" value="-Xmn1024m -Xgcpolicy:gencon"/>	

Figure 3. FileNet P8 Content Engine - JVM heap and argument settings

-Xmn1024m -Xgcpolicy:gencon

where

- Xmn set the size of new generation of Java heap space
- Xgcpolicy Parameter to change the garbage collection mechanism

WAS administration console path:

Servers > Application Servers > server1 > Java and process management > Process Definition > Java Virtual Machine

WAS JVM logging

Establish log file rotation configuration for **System.out** and **System.err** log files to be self-managing. The log history was set to 50 MiB in total for the SUT.

The screenshot shows the 'Log File Rotation' configuration window. It has two main sections. The first section has two radio buttons: 'File Size' (checked) and 'Time' (unchecked). Under 'File Size', there is a 'Maximum Size' label and a text input field containing '10', followed by 'MB'. Under 'Time', there is a 'Start Time' label and a text input field containing '24', and a 'Repeat Time' label with a text input field containing '24' followed by 'hours'. The second section is labeled 'Maximum Number of Historical Log Files. Number in range 1 through 200.' and has a text input field containing '5'.

Figure 4. JVM log file rotation settings

WAS administration console path:

Troubleshooting > Logs and Trace > server1 > JVM Logs

WAS IBM service logs

IBM service logs were **disabled**.

WAS administration console path:

Troubleshooting > Logs and Trace > server1 > IBM service logs

WAS Performance Monitoring Infrastructure (PMI)

The Performance Monitoring Infrastructure (PMI) has been **disabled**.

WAS administration console path:

Monitoring and Tuning > Performance Monitoring Infrastructure (PMI) > server1

JDBC settings

The minimum/maximum Connections was set to **100** for the object store database and index JDBC data sources.

WAS administration console path:

Resources > JDBC > Data sources > 'Data source name' > Connection pools

The following custom properties were explicitly set for the object store database and the global configuration database (GCD) JDBC data sources.

- ResultSetHoldability to **1**
- WebSphereDefaultIsolationLevel to **2**

WAS administration console path:

Resources > JDBC > Data sources > 'Data source name' > Custom properties

Transaction lifetime timeout

The total transaction lifetime timeout was set to **600**.

WAS administration console path:

Servers > Application Servers > server1 > Container Services > Transaction service

Configure the Object Request Broker (ORB)

The maximum size for the thread pool was set to **100** threads.

Application servers > server1 > ORB service > Thread Pool

Use this page to specify a thread pool for the server to use. A thread pool enables server components to reuse threads instead of creating new threads at run time. Creating new threads is typically a time and resource intensive operation.

Configuration

General Properties

- * Minimum Size
 threads
- * Maximum Size
 threads
- * Thread inactivity timeout
 milliseconds

Allow thread allocation beyond maximum thread size

Figure 5. ORB thread pool properties

WAS administration console path:

Servers > Application Servers > server1 > Container Services > ORB service > Thread pool

The ORB custom property *com.ibm.CORBA.FragmentSize* was set to **0**.

WAS administration console path:

Servers > Application Servers > server1 > Container Services > ORB service > Custom properties

For more information on FileNet P8 performance tuning, refer the FileNet P8 Version 5.1 Information Center:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

Administering IBM FileNet P8 > Performance tuning IBM FileNet P8 components

FileNet P8 Process Engine (PE)

After installing and configuring the FileNet P8 PE application you can verify that the PE installation was successful by accessing the FileNet P8 PE server information (ping page).

Browse the FileNet P8 PE ping page:

`http://<server>:<port>/IOR/Ping`

Note: The default PE naming service port is **32776**.

For more information, refer the FileNet P8 Version 5.1 Information Center:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

Installing or upgrading IBM FileNet P8 Platform > Installing a distributed IBM FileNet P8 Platform system > Installing and configuring Process Engine > Verifying Process Engine installation

Tuning for the FileNet P8 Process Engine (PE)

PE Java Virtual Machine settings

The PE Java Virtual Machine heap size was set to **1024 MiB** initial/maximum heap size. The PE heap size can be changed by editing the PE application startup script:

`/opt/IBM/FileNet/ProcessEngine/startpesvr`

The PE Java startup command line was then changed as follows:

```
$JAVA_HOME/bin/java -Djpeserver.useHTTPTunneling=false -Djpeserver.httpunnel.ssl=false
-Xms1024m -Xmx1024m -Xmn512m -Xgcpolicy:gencon
-cp
$JPEINSTALL_DIR/lib/pesvr.jar:$JDBC_JAR_PATH:$JPEINSTALL_DIR/CE_API/lib/Jace.jar:$JPEINSTALL_DIR
/CE_API/wsi/lib/wasp.jar -DJPEINSTALL_DIR=$JPEINSTALL_DIR -DJPEDATA_DIR=$JPEDATA_DIR -
DJPEVIRTUALSERVER=$JPEVIRTUALSERVER -
Djava.security.auth.login.config=$JPEINSTALL_DIR/CE_API/config/jaas.conf.WSI -
Dwasp.location=$JPEINSTALL_DIR/CE_API/wsi filenet.jpe.rpc.main.PEServer
```

where

-Xms	set the initial Java heap size
-Xmx	set the maximum Java heap size
-Xmn	set the size of new generation of Java heap space
-Xgcpolicy	Parameter to change the garbage collection mechanism

DB2 CE/PE database server

The IBM DB2 documentation describes the requirements for a Linux DB2 installation. For example, the *libaio* package is required for DB2 database servers using asynchronous I/O.

For more information on DB2 installation requirements, refer the IBM DB2 Database for Linux Information Center:

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/index.jsp>

Database fundamentals > Installing > Database systems > DB2 Database for Linux and UNIX > DB2 Servers > Installation prerequisites (Linux and UNIX)

Note: Starting with IBM DB2 Enterprise Server Edition Version 9.5 Fix Pack 6 the database manager uses a new formula for automatic kernel parameter adjustments. For earlier fix pack versions, you must manually update the kernel parameter settings.

Tuning the DB2 CE/PE database server

Set the DB2 workload type

Since DB2 version 9.7 (and DB2 version 9.5 fix pack 5) FILENET_CM is introduced as a possible value for the DB2_WORKLOAD registry parameter. When setting the DB2_WORKLOAD variable to FILENET_CM it implicitly sets DB2_EVALUNCOMMITTED and DB2_SKIPINSERTED on behalf.

Sample command: db2set setting DB2_WORKLOAD to FILENET_CM

```
# db2set DB2_WORKLOAD=FILENET_CM
```

See also in the FileNet P8 Version 5.1 Information Center:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

Administering IBM FileNet P8 > Performance tuning IBM FileNet P8 components > Tuning IBM FileNet P8 databases > Tuning an IBM DB2 database for IBM FileNet P8 components

Enable DB2 parallel I/O

DB2 parallel I/O was enabled for all CE/PE data tablespaces. The tablespaces for the object store database and the global configuration database (GCD) are located on a Linux Logical Volume (LV).

See also chapter 'Linux setup - Disk I/O – Using the Logical Volume Manager'

Sample command: db2set enabling DB2 parallel I/O for tablespaces

```
# db2set DB2_PARALLEL_IO=*:4
```

The above command enables DB2 parallel I/O for all tablespaces. The underlying Logical Volumes (LV) for the database data and log had 4 physical DASDs each. DB2 uses this registry variable to change the way it calculates the I/O parallelism for tablespaces.

Sample command: db2set displaying the registry variable settings

```
# db2set
DB2_WORKLOAD=FILENET_CM
DB2_SKIPINSERTED=YES [DB2_WORKLOAD]
DB2_EVALUNCOMMITTED=YES [DB2_WORKLOAD]
DB2COMM=tcPIP
DB2_PARALLEL_IO=*:4
```

See also in the FileNet P8 Version 5.1 Information Center:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

Administering IBM FileNet P8 > Performance tuning IBM FileNet P8 components > Tuning IBM FileNet P8 Content Engine > Tuning IBM FileNet P8 Content Engine databases

LDAP Security Server

One z/VM guest was setup as a LDAP (Lightweight Directory Access Protocol) security server. IBM FileNet P8 supports the IBM Tivoli Directory Server for providing directory services amongst others.

The security server does not require much resources and no special tuning was applied for this guest. The single virtual CPU defined for this guest was sufficient for this SUT and the workload scenarios.

For more information, refer the FileNet P8 Version 5.1 Information Center:

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

Security > IBM FileNet P8 security > Directory service providers > IBM Tivoli Directory Server

Workload Description

Testing was conducted using a load generation application developed by IBM on an IBM Flex System client machine. A combined workload mix of Content and Process Engine operations were performed simultaneously in an attempt to accurately depict a possible customer scenario. It is referenced as ECM workload driver in throughout the paper.

CE Java API:

The system was pre-populated with over one million documents ranging in size from 1 KiB to 5 MiB, in classes ranging from 5 to 50 custom properties, and over 200K folders, and 60K custom objects. About half of the documents used database storage and half filestore storage.

The workload is made up of a realistic mix of the most common CE Java API transactions, with an overall ratio of 70% queries and retrieves and 30% creates and updates. The following table lists the detailed workload shares:

CE transactions	Workload share
<u>Authoring</u>	
• create documents, ranging from 1KiB to 500KiB	16%
• update documents of 100KiB	14%
<u>Retrieval</u>	
• retrieve documents, ranging from 1KiB to 2MiB	36%
<u>Search and foldering</u>	
	34%

Table 8.1. CE workload mix

The CE transactions access data in the Linux filesystem CE file store and the CE object store on the database server.

PE Java API:

The system was pre-populated with 500,000 work items in 3 different queues. The workload is made up of a realistic mix of the most common PE Java API transactions. PE transactions access data in the PE workflow database on the database server.

PE transactions	Workload share
Logon	5%
Query	20%
Update	30%
Dispatch	30%
Reassign	5%
Launch	10%

Table 8.2. PE workload mix

Results

The CPU scalability and workload scaling runs with the benchmark application are based on a typical customer like workload transaction mix. The results of the benchmark runs are discussed in this chapter. The transaction mix consists of typical PE and CE transactions at the same time. The results section is divided in a workload scaling and a CPU vertical scalability part.

The following table explains the terms and labels used in the result charts.

Chart terms	Description
increasing workload level	The ECM workload driver can be adjusted to generate a desired workload level. This can be either done by adding more users or reducing the user inactivity time (sleep time) for the users.
normalized throughput	The actual achieved normalized throughput rate for the adjusted workload level. The normalized throughput rates vary from 1 (lowest) to 35 (highest) in the charts.
average response time [ms]	The average response time for CE or PE transactions measured in milliseconds (ms).
CPU load [IFLs used]	The Integrated Facility for Linux (IFL) is an IBM mainframe processor dedicated to the Linux operating system, even when running under z/VM. 'IFLs used' is the number of physical processors consumed for a certain CPU load.
CE – Create/Update/Retrieval	CE operation types (file store and database): <u>Create</u> : create, delete documents of different sizes <u>Update</u> : check out-in, update documents of different sizes <u>Retrieval</u> : retrieve, query, get documents of different sizes
PE – Query/Reassign/Update/Dispatch/ Launch/Logon	PE operation types: <u>Query</u> : searches queues, work items, work objects and rosters <u>Reassign</u> : reassign work items <u>Update</u> : updates work items and work objects <u>Dispatch</u> : dispatches work items and work objects <u>Launch</u> : launches workflows <u>Logon</u> : logon PE users

Table 9. terms used in the charts

Workload scaling

One exemplary ECM workload scaling measurement series is discussed here. The ECM workload was increased until all available CPUs on the CE/PE guest were fully utilized. The number of CPUs on the CE/PE guest is considered as the limiting factor for the SUT at the highest possible throughput rate for a scenario.

The measurement series for the other CE/PE guest CPU configurations behave similarly at lower (for 1 CPU) or accordingly higher (for 4 and 8 CPUs) throughput rates. The 2 CPU scenario is discussed in

the following section. The other measurements series will be described in the vertical CPU scalability discussion later on.

CE/PE throughput and CPU load

The CE/PE and the DB2 database server were defined with 2 virtual guest CPUs each for this scenario. The ECM workload was increased until the 2 CPUs on the CE/PE guest are fully used.

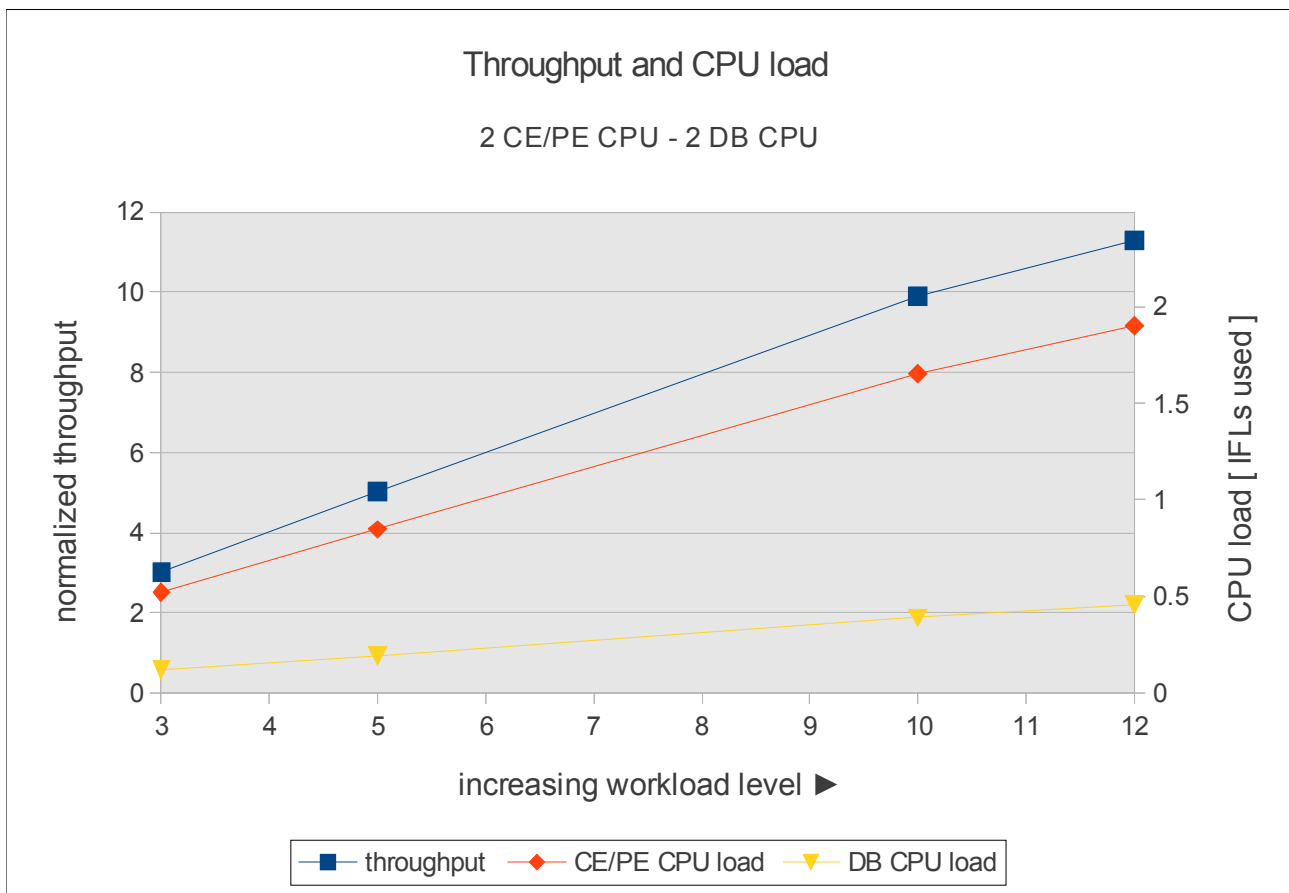


Figure 6. Transaction throughput and CPU load

The figure above shows the normalized transaction throughput for different workload levels and the CPU loads on the CE/PE and DB server. A throughput rate of '1' corresponds to the transaction throughput for the lowest workload level with one CE/PE guest CPU.

The blue (dark) line in chart shows the CE/PE transaction throughput for the increasing workload. The red and yellow (light) lines show the corresponding CPU load on the CE/PE and the database (DB) guest.

Observations

The workload was increased until almost 2 CPUs on the CE/PE guest are fully used. The load on the database server reaches 0.5 IFLs at the highest workload level. The resulting transaction throughput nearly quadruples at the highest workload level compared to the lowest workload level.

The throughput scales linearly until the CE/PE CPUs are fully used, resulting in a similar trend line for the throughput and CE/PE CPU load. This is a very good scaling behaviour, only the number of CPUs on the CE/PE guest limits the transaction throughput for the users. There is no indicator for any overhead at the higher workload levels, because the CPU load follows exactly the throughput and does not increase faster.

CE average response times

The next figure lists the average response times for CE operations at increasing workload levels. All single CE operations are grouped into three categories (Create / Update / Retrieval) and the response times are an average value for these groups. For this particular workload using direct CE Java API calls, response times below 200 ms are expected and good.

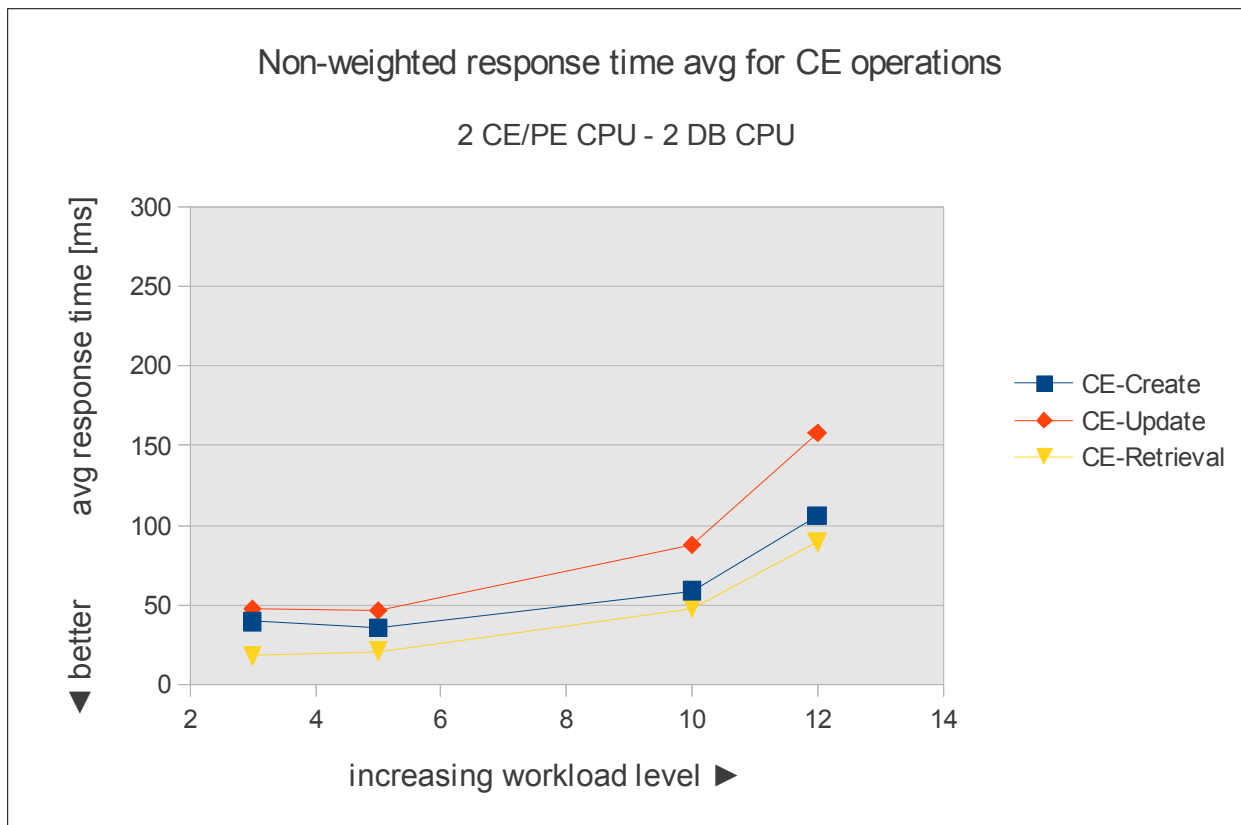


Figure 7. Non-weighted average response times for CE operations

Observations

The average response times are below 50 ms for the lower and below 100 ms for the higher workload levels. The rightmost highest workload level shows a higher increase in response times, but still below 200 ms, showing continued good performance up to nearly full system utilization.

PE average response times

The next figure lists the average response times for PE operations at different workload levels. All PE operations are grouped into six categories (Query / Reassign / Update / Dispatch / Launch / Logon) and the response times are an average value for these groups. For this particular workload using direct PE Java API calls, response times below 200 ms are expected and good.

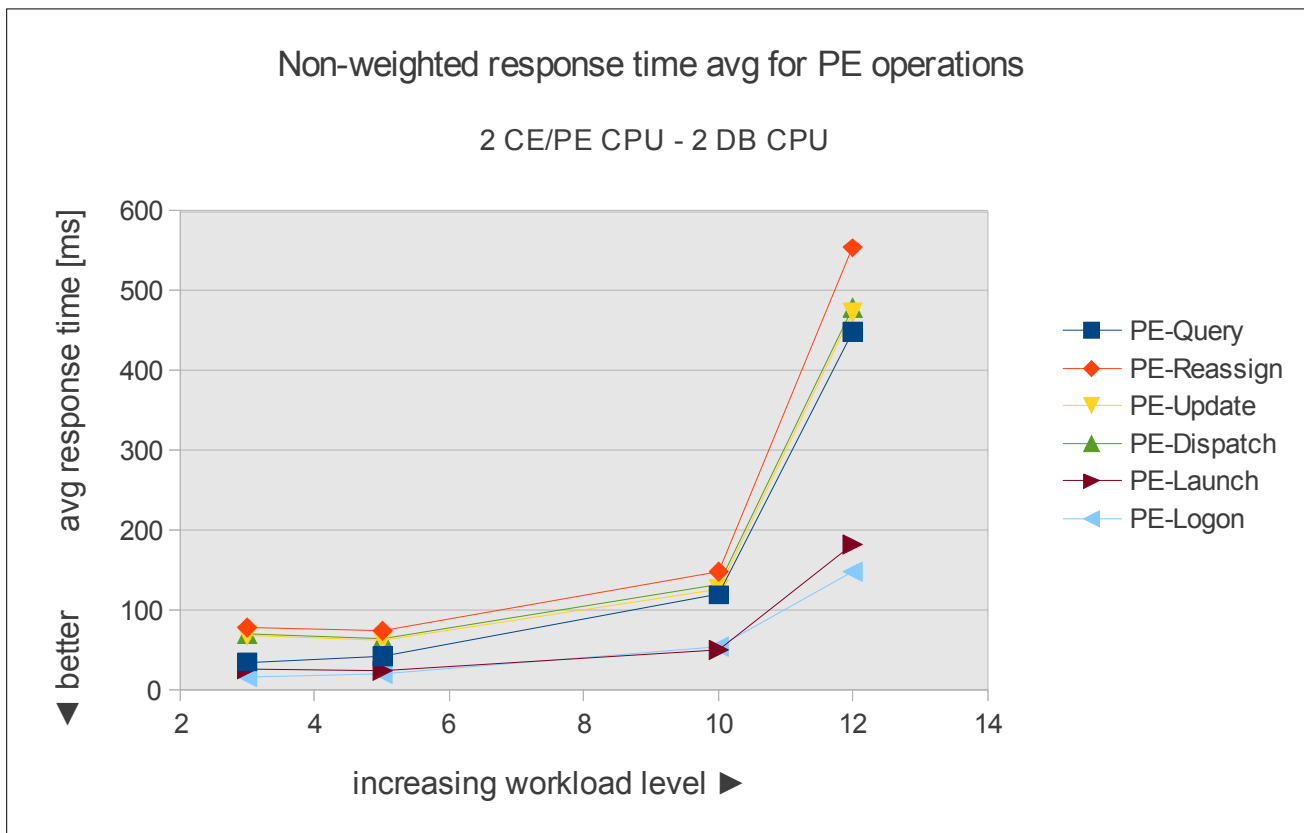


Figure 8. Non-weighted average response times for PE operations

Observations

The average response times are stable and low over most of the workload range. At the highest workload level (above 10), at the CE/PE guest CPUs near maximum utilization, PE API response times also start to increase (while remaining well below 1 second). Close to the maximum CPU utilization

processes on the CE/PE server are waiting on CPU power which results in higher response times.

Adding further virtual CPUs reduced response times for both CE and PE operations, allowing scaling to higher workload levels with good response times, as demonstrated in the following section.

CE/PE vertical scalability

To demonstrate the vertical scalability of the FileNet P8 CE/PE server running under Linux for System z, the number of virtual CPUs was increased for the CE/PE and database server guests. The number of virtual CPUs for the CE/PE guest varied from 1 to 8 CPUs, and the number of virtual CPUs for the database server varied from 1 to 2 CPUs depending on the scenario. Two virtual CPUs for the database server were enough to handle even the highest workload rates for these tests.

The ECM workload was adjusted for every single CPU configuration, so that the CPUs on the CE/PE guest are busy with the highest possible throughput rate. The next table lists the CPU configurations and workload levels that have been chosen for the CPU scaling measurement series. A throughput rate of '1' corresponds to the transaction throughput for the lowest workload level with one CE/PE guest CPU.

z/VM Guest CPU configurations	maximum throughput rates (normalized)
1 CE/PE CPU – 1 DB CPU	1 to 5
2 CE/PE CPU – 2 DB CPU	up to 11
4 CE/PE CPU – 2 DB CPU	up to 22
8 CE/PE CPU – 2 DB CPU	up to 39

Table 10. CPU scaling scenarios and maximum throughput rates

CE/PE throughput and CPU load

The measurement series with the maximum throughput number for each CE/PE guest CPU configuration was selected for the vertical scalability chart below. Besides the throughput bars, two trend lines indicate the CPU load for the CE/PE and the database guest belonging to the throughput rates. The bars for the normalized transaction throughput are arranged accordingly to their workload level in the chart.

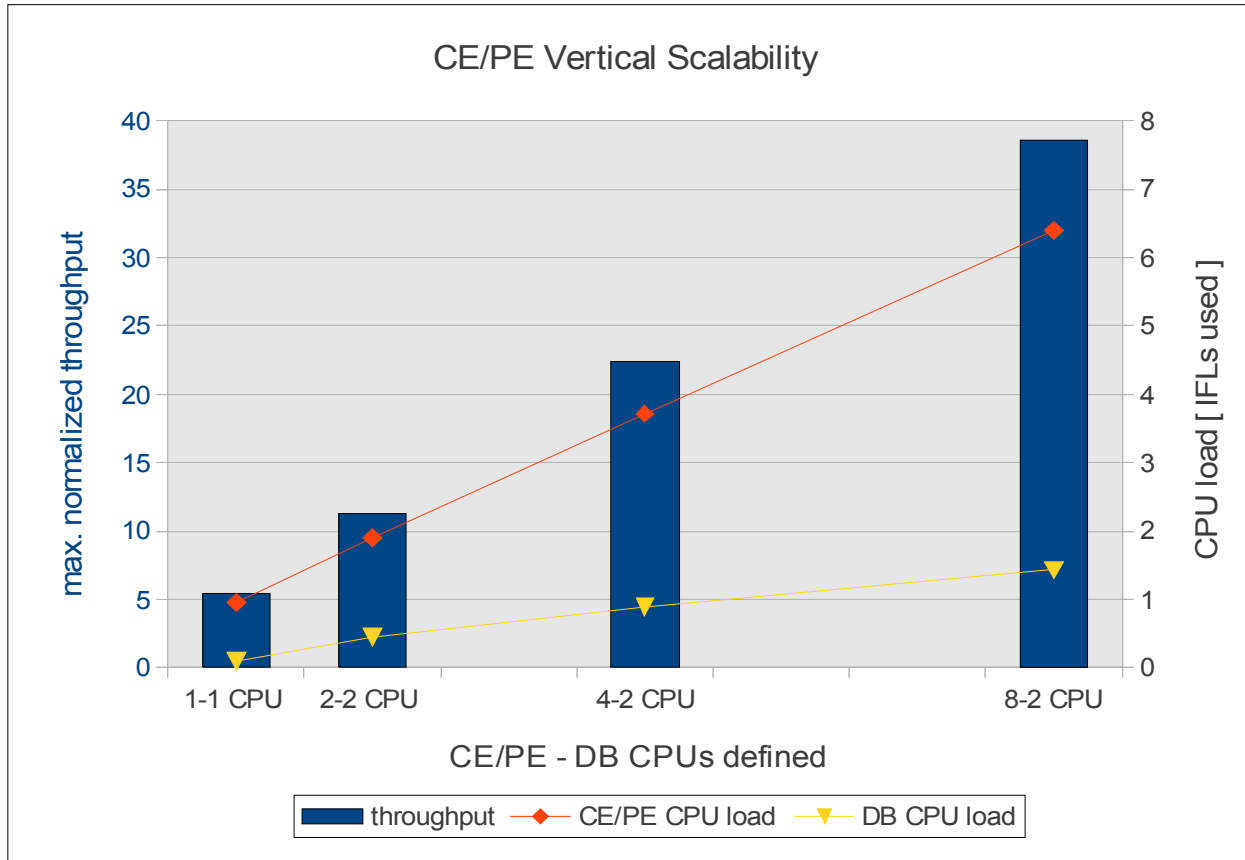


Figure 9. CE/PE vertical scalability - transaction throughput and CPU load

The blue (dark) bars in chart show the maximum transaction throughput for all CPU configurations at increasing workload levels. The red and yellow (light) trend lines show the corresponding CPU load on the CE/PE and the DB (database) guest.

Observations

The maximum throughput rates start from 5 for the 1-1 CPU scenario to nearly 40 for the 8-2 CPU scenario. To drive each CPU scenario with the maximum workload almost all CPUs on the CE/PE guest are used. On the database server the CPU load grows smoothly from 0.1 to 1.5 IFLs used for the highest throughput rate at the 8-2 CPU scenario.

Each doubling of the CPUs on the CE/PE guest allowed us to achieve a nearly doubled workload. The throughput rate increase shows a nearly perfect scalability characteristic for the full workload bandwidth (see also Figure 9). Adding more CPUs to the CE/PE guest allows to drive higher throughput rates when the server is already running CPU bound. The CPU load on the database server should also be monitored, and in our scenarios the database server CPU load grows smoothly and not as fast as on the CE/PE server for the chosen ECM workload.

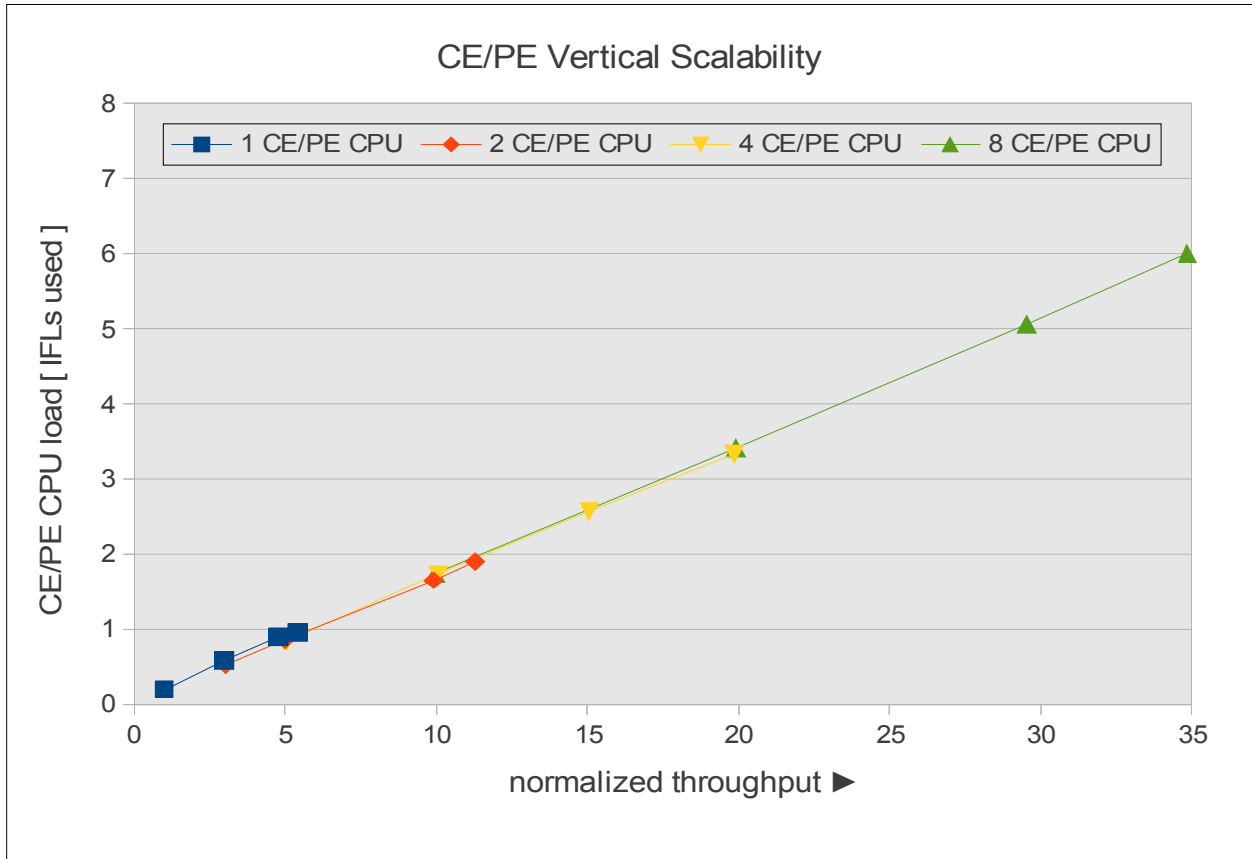


Figure 10. CE/PE vertical scalability – CE/PE CPU load for transaction throughput rates

These tests illustrate the good vertical scalability characteristics for FileNet P8 on Linux for System z when increasing the number of CPUs for the scenarios and the workload for the CE/PE guest. The linear scalability range ranges from 1 used IFL through 6 IFLs, demonstrating very nice symmetric multiprocessing (SMP) performance, regardless how much virtual CPUs are configured for the CP/PE guest the used CPU capacity is the same. There is no additional CPU cost, just because there are more CPUs available.

CE average response times

For considering the CE average response times across the different CE/PE guest CPU scenarios a single average response time for all individual CE transaction types has been calculated. This value is an arithmetic mean across all CE operation categories (Create / Update / Retrieval). The next chart shows these overall average response times for the CPU scenarios in relation to their throughput levels.

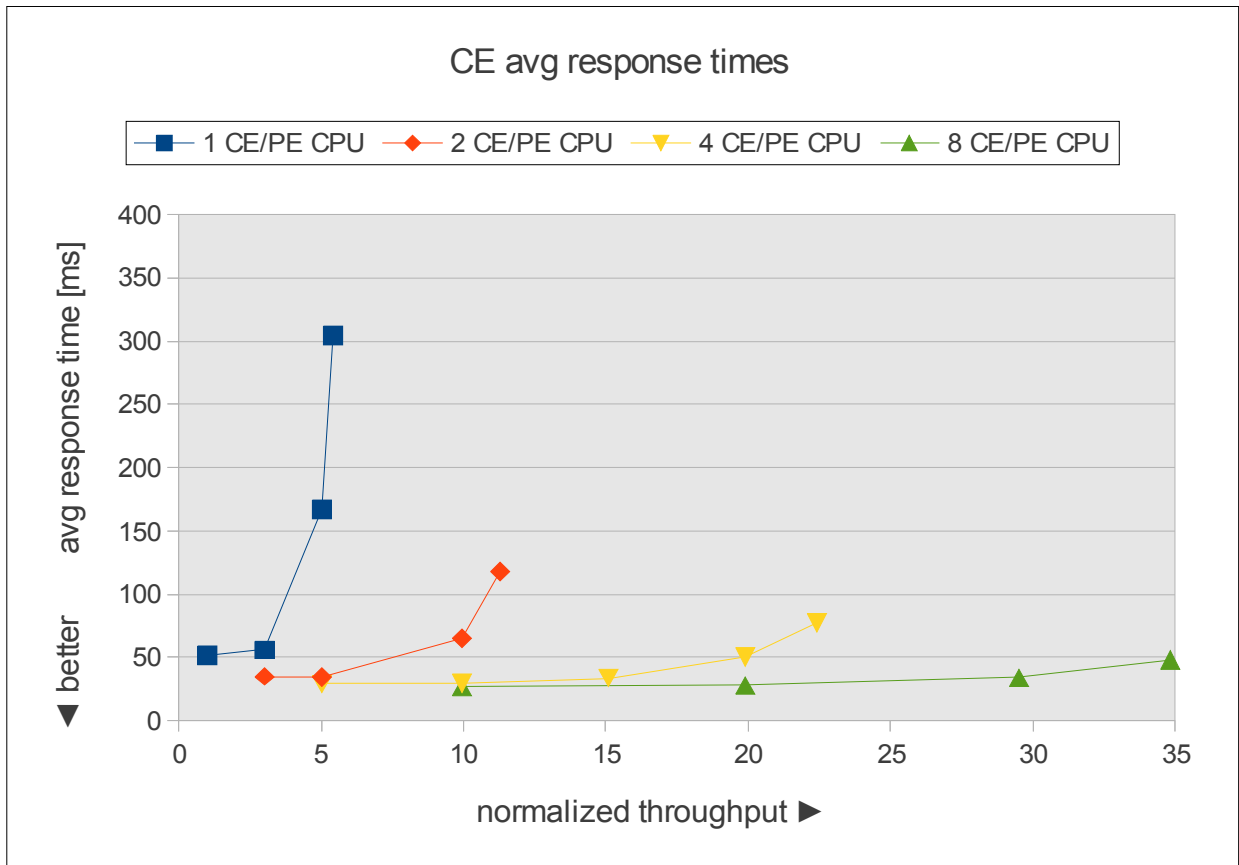


Figure 11. Non-weighted average response times for CE operations in relation to the throughput numbers and CPUs used

Observations

For this particular workload using direct Java API calls, response times below 200 milliseconds are expected and good. The average response times measured are stable and low over most of the workload range. At the very highest workload level, as the CE/PE guest CPUs near maximum utilization, API response times also start to increase (while remaining well below 1 second).

The CE average response times become much better the more CPUs are available on the CE/PE guest. The best response times are achieved with 8 CPUs. In the workload range where the CPUs are limiting the throughput for a certain scenario (usually greater than 80% CPU load), the response times start to increase. Adding further CPUs to such a CPU bound system takes the response times down again and allows further throughput increase. Having in mind that more virtual CPUs on the CE/PE guest for the same workload level are not leading to additional CPU overhead (see figure 9) but improving the response times, seem to indicate that the system benefits from a higher degree of parallelism.

PE average response times

For considering the PE average response times across the different CE/PE guest CPU scenarios a single average response time for all individual PE transaction types has been calculated. This value is an arithmetic mean across all PE operation categories (Query / Reassign / Update / Dispatch / Launch / Logon). The next chart shows these overall average response times for the CPU scenarios in relation to their throughput levels.

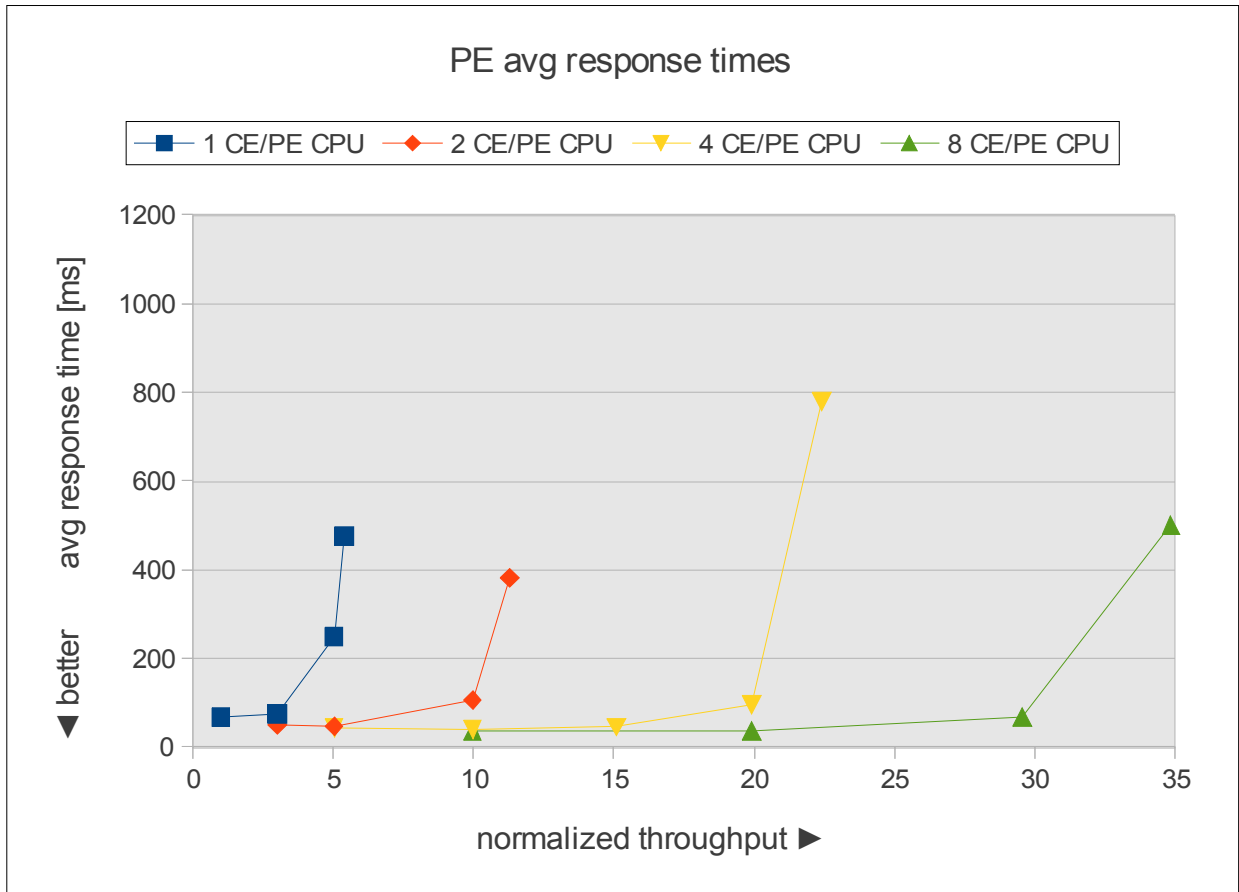


Figure 12. Non-weighted average response times for PE operations in relation to the throughput numbers and CPUs used

Observations

For this particular workload using direct Java API calls, response times below 200 milliseconds are expected and good. The average response times measured are stable and low over most of the workload range. At the very highest workload level, as the CE/PE guest CPUs near maximum utilization, API response times also start to increase (while remaining well below 1 second).

Similarly to the CE response times, the PE average response times remain low until a certain CPU usage level is reached (usually above 80% CPU load). Adding more CPUs for these workload levels takes the response times down again and allows further throughput increase.

The amount of CPUs on the database system has a minor role, two were sufficient for all scenarios, for the lowest workload levels with one CE/PE CPU only one CPU was sufficient.

Appendix

The following tables give a brief overview of the tuning steps done for z/VM, Linux on System z, DB2 and for FileNet P8 CE/PE. All tuning and setup steps are outlined in detail in SUT setup chapters.

This appendix is not intended as a cookbook of tunings to apply to every FileNet P8 deployment, but it does describe the tunings that were done in this particular environment for this particular workload. For performance tuning of an actual FileNet P8 deployment, use the performance tuning resources noted in the References.

z/VM area	setting
guest-to-guest networking	VSWITCH defined
disk I/O	HyperPAV aliases directly attached to disk I/O intensive guests
memory management	turned off reorder processing for guests with \geq 8GiB memory

Table 11: z/VM adaptations

Linux area	setting
guest-to-guest networking	VSWITCH deployed
disk I/O	HyperPAV used for LVM DASDs on the disk I/O intensive guests
maximum number of open files on the WAS server	set to 8192 (ulimit)

Table 12: Linux adaptations

DB2 area	setting
DB2_WORKLOAD registry parameter	set to FILENET_CM db2set DB2_WORKLOAD= FILENET_CM
DB2 parallel I/O for table spaces	db2set DB2_PARALLEL_IO=*:[number of disks per tablespace container] set to 4 for the SUT outlined in this paper

Table 13: DB2 adaptations

FileNet P8 CE area	setting
WAS JVM initial/maximum heap size	set to 2048 MiB
WAS generic JVM arguments	-Xmn1024m -Xgcpolicy:gencon
WAS JVM logging	enabled log rotation log history set to 5x10MiB
WAS IBM service logs	disabled
WAS Performance Monitoring Infrastructure (PMI)	disabled
JDBC settings	minimum/maximum Connections was set to 100 for the object store database and index JDBC data sources set custom properties for the object store database and the global configuration database (GCD) JDBC data sources : - ResultSetHoldability to 1 - webSphereDefaultIsolationLevel to 2
WAS container services	set total transaction lifetime timeout to 600
WAS Object Request Broker (ORB)	set the maximum size for the thread pool to 100 threads set custom property com.ibm.CORBA.FragmentSize to 0

Table 14: FileNet P8 CE adaptations

FileNet P8 PE area	setting
PE JVM initial/maximum heap size	set to 1024 MiB
generic JVM arguments	-Xmn512m -Xgcpolicy:gencon

Table 15: FileNet P8 PE adaptations

References

ECM products for System z

<http://www.ibm.com/software/data/content-management/systemz/>

How to Improve Performance with PAV, SC33-8414

http://www.ibm.com/developerworks/linux/linux390/development_documentation.html

Oracle Database on Linux on System z - Disk I/O Connectivity Study

http://www.ibm.com/developerworks/linux/linux390/perf/tuning_database.html#con

Linux on System z - Device Drivers, Features, and Commands SC33-8411-13

<http://www.ibm.com/support/docview.wss?uid=pub1sc33841113>

Details about Linux for System z network tuning

http://www.ibm.com/developerworks/linux/linux390/perf/tuning_networking.html

IBM FileNet P8 Version 5.1 Information Center

<http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/index.jsp>

IBM WebSphere Application Server Version 7.0 Information Center (online)

<http://pic.dhe.ibm.com/infocenter/wasinfo/v7r0/index.jsp>

IBM DB2 Database for Linux, UNIX, and Windows Information Center (online)

<http://publib.boulder.ibm.com/infocenter/db2luw/v9r5/index.jsp>

IBM Tivoli Directory Server version 6.3

<http://pic.dhe.ibm.com/infocenter/tivihelp/v2r1/index.jsp>

Input/Output Configuration Program User's Guide, SB10-7037-10

<http://www.ibm.com/servers/resourceink>

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:
IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Software Interoperability Coordinator, Department 49XA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.



©Copyright IBM Corporation 2013
IBM Systems and Technology Group
Route 100
Somers, New York 10589
U.S.A.
Produced in the United States of America,
08/2013

IBM, IBM logo, DB2, ECKD, FileNet, Flex System, System Storage, System x, System z, Tivoli, WebSphere, zEnterprise, and z/VM are trademarks or registered trademarks of the International Business Machines Corporation.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license there from.

InfiniBand and InfiniBand Trade Association are registered trademarks of the InfiniBand Trade Association.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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

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

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

This document is provided "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranty of merchantability or fitness for a particular purpose. This document is intended for informational purposes only. It could include technical inaccuracies or typographical errors. The information herein and any conclusions drawn from it are subject to change without notice. Many factors have contributed to the results described herein and IBM does not guarantee comparable results. Performance numbers will vary greatly depending upon system configuration. All data in this document pertains only to the specific test configuration and specific releases of the software described.