École Publique d'Ingénieurs de la Santé et du Numérique
Jointly with Groupe BPCE

Mainframe footprint and consumption, an evaluation based on sustainable IT approach

Apprenticeship supervisor
[redacted]

Academic supervisor
Olivier JOFFRE

Pierre JACQUET
EPISEN, SI Faculty
2020-2021 graduation year

Mainframe footprint and consumption, an evaluation based on sustainable IT approach

Apprenticeship supervisor
[redacted]

Academic supervisor
Olivier Joffre

Pierre Jacquet

École Publique d'Ingénieurs de la Santé et du Numérique,
Information System (SI) Faculty,
Jointly with Groupe BPCE

2020-2021 graduation year

Acknowledgements

I would like to thank all the people who made this document possible. First of all, I particularly thank my supervisor [redacted] for his trust and his investment in my apprenticeship. I am grateful to my colleagues at BPCE, who welcomed me three years ago and taught me everything I had to know about the mainframe world : [redacted] I also benefited from the help of [redacted] who assisted me with most of the elements of the mainframe architecture. Finally, I thank my academic supervisor, Olivier Joffre, for his follow-up during the last five years.

Abstract

Information and communications technology (ICT) must be integrated into a sustainable approach due to their strong growth, which generates an increasingly significant carbon footprint. To apply a digital sobriety approach, we started by studying the possible variations in power consumption valuation methods to a mainframe server. After having justified that the software valuation would be the most appropriate, we demonstrated by a practical study that it was not sufficient given the high static consumption of a mainframe server and the consumption of the related non-CPU external components. By estimating the overall consumption and carbon footprint of Groupe BPCE's mainframe platform, we justified that they were similar to an x86 platform with the same availability constraints. Finally, we detailed the specificities of the mainframe ecosystem and its possible evolutions to justify what a sustainable mainframe platform would be.

Résumé

La nécessité d'intégrer le numérique dans une démarche durable vient notamment de sa forte croissance, engendrant une empreinte carbone de plus en plus importante. Pour appliquer une démarche de sobriété numérique, nous commençons par étudier les déclinaisons possibles des différentes méthodes d'évaluation de la consommation électrique d'un serveur au mainframe. Après avoir justifié que l'évaluation logicielle serait la plus appropriée, nous démontrons par l'étude pratique qu'elle n'est pas cependant suffisante vu la forte consommation statique d'un serveur mainframe et vu la consommation des composants hors-CPU connexes. En estimant la consommation globale et l'empreinte carbone de la plateforme mainframe du Groupe BPCE, nous justifions qu'elles sont similaires à une plateforme x86 distribuée avec les mêmes contraintes de disponibilité. Finalement, nous nous intéressons aux spécificités de l'écosystème mainframe et à ces possibles évolutions afin de justifier ce que serait une plateforme mainframe durable.

Contents

Chapter 1

Introduction

According to IBM, 92 of the top 100 banks use mainframe computers and 90% of all credit card transactions are performed on mainframes. They are a key architecture element for many companies such as Groupe BPCE, which is one of the main French banking groups with multiple subsidiary companies like Banques Populaires, Caisses d'Epargne and Natixis.

Information and communication technologies are sometimes seen as a way to reduce carbon footprint[01]. However, based on the rapid growth of these technologies, they may also be an aggravating factor of climate change based on their greenhouse gas emissions[02]. By 2025 they could account for 20% of the global energy use globally[03]. Therefore, researchers have detailed ways to reduce consumption and carbon footprint based on both hardware and software approaches.

These approaches are usually based on common platforms, such as x86 computers, and there are, to the best of our knowledge, no dedicated studies to the mainframe ecosystem with a consumption prism, despite its importance in many Information systems. We believe that mainframe platforms should be studied more deeply to improve banking energy efficiency and to determine ways to reduce a financial transaction carbon footprint. In consequence, we propose to study a mainframe platform based on its consumption and carbon footprint elements.

As sustainable ICT depends on accurate metrics; we first study consumption evaluation methods and how to decline them on a mainframe platform in regards to its specificities. Hardware, hybrid, and software approaches are compared with mainframe architecture.

In the second part, we determine the Groupe BPCE mainframe server's consumption and carbon footprint. To complete it, we also study external components' impact, such as tape libraries and flash storage arrays on the global platform, and therefore conclude with overall values. As a way to compare the given order of magnitude, we evaluate an equivalent x86 architecture and compare its consumption and carbon footprint.

Finally, we study how mainframe specificities could be associated with a sustainable strategy based on state of art optimization, but also based on its architecture and retro compatibility capacity. Its future evolution is estimated using some identified internal and external factors and we conclude by a lead of what could be a sustainable mainframe strategy.

Chapter 2

Consumption evaluation methods applied to mainframes

Evaluating the consumption of a given software has multiple approaches depending on the platform studied. The paper introduces mainframe specificities in section 2.1 and then details consumption evaluation methodologies that could be applied to sections 2.2 and 2.3.

2.1 Mainframe elements

Mainframes are computers designed and used for specifics use-cases. To evaluate their consumption properly, we start by describing the global architecture of a mainframe platform, before detailing the type of work running on it. We finally list some existing monitoring facilities.

2.1.1 Mainframe architecture

Mainframe computing is a paradigm based on resource centralization[04]. It was created in the late 1950s for company needs with the technical limits of first computers: components size, cost, and restricted interoperability. The mainframe unit contains the CPU, the RAIM (Redundant Array of Independent Memory), the software, and is connected through its input/output to multiple devices such as storage servers, printers, and tape drives. Database operation, transaction, and batch processing are operated on the same unit.

Mainframes have been evolving for more than 60 years, leading to significant changes in architecture. It remains, however, a computer with a high amount of input/output channels, a high availability (dynamic capabilities for upgrading hardware and software, hardware error recovery automation, redundancy), and a shared memory architecture (up to 40To of RAIM and 190 cores at 5.2GHz in the latest models). This paper is based on the z15 model T01, which is the newest version of IBM as of 2021. The machine will hereafter be referred to as z15.

The current IBM architecture is usually based on Logical Partitions (LPAR). Mainframes host at least one LPAR but typical mainframe environments host multiple LPARs for development and test purpose. LPAR configuration is performed using the hypervisor Processor Resource/System Manager (PR/SM). Each LPAR of the platform has its own independent operating system (OS). A compatible OS includes z/OS, Linux, AIX, z/VM, and z/TPF. The traditional mainframe OS is z/OS, a successor of MVS. This paper is based on its use.

Large companies tend to use mainframes clusters because of the limit on the number of LPARs per machine, but also for higher availability (enhancement of disaster recovery) and higher performance. Using IBM Parallel Sysplex, a production LPAR is usually shared between physical distinct mainframes.
Typical mainframe software includes Time Sharing Option (TSO) and Transaction Processing Monitors (TP Monitor). TSO is an environment mainly used by system administrators and developers to interact with the computer system. It provides a text editor, batch job support, script language support (REXX, CLIST), debuggers, and interfaces with other vendors' software. TP Monitors manage business transactions by displaying a potential graphic user interface (GUI), receiving data, and updating databases in a single transaction. They are mostly called, directly or not, by users.

Operators access mainframes using 3270 terminals, which used to be physical 3270 connections but are now emulated with software clients or web interfaces. However, the current dynamics seem to limit 3270 connections to system administrators and developers. The historical GUIs proposed to users by TP Monitors have progressively been replaced by web applications or thick clients which call TP Monitors using connectors.

The current mainframe ecosystem is evolving with two different technologies: Linux, and containers. Since 1998 an interface with the Unix world is available on z/OS with the Unix System Services (USS). IBM also proposes to run Linux directly on a traditional mainframe LPAR or dedicated mainframe unit to Linux, called LinuxOne. Containers are supported by z/OS 2.4, and OpenShift orchestration support is announced. Both technologies are disruptive and could change the use of mainframes. They are however not currently used in Groupe BPCE and could not be studied in this paper.

2.1.2 Type of mainframe work

Due to their architecture and longevity, mainframes have a specific place in the information system. They usually host legacy applications written in Cobol, Fortran, PL/1, and Assembler. Other languages, such as Java and C/C++, can run on mainframe platforms but are less common. Mainframe programs can be critical to the organization, using the platform's ability to manage high numbers of simple calculations.

Mainframe applications are mostly related to transaction processing. A transaction can be defined as an activity that changed stored data[05]. It is a unit that contains different operations which can succeed, or fail, as a whole. A failure will let the previous data unchanged, whereas a success will write a new context.

A transaction can be performed as a batch. Batch processing uses gathered information to act at a specific time which is usually orchestrated by a scheduler such as Tivoli Workload Scheduler (TWS). Mainframe systems used to be oriented toward batch processing, leading to multiples facilities such as the Job Control Language (JCL). This language describes the job identity, the required libraries, input, output, and the different programs to execute. Each batch is described using JCL and submitted to the Job Entry Subsystem (JES). JES manages batch execution and optimizes job processing using different mechanisms, such as spooling common storage volumes between jobs, or parallelism between jobs steps.
A transaction can also be performed in real time. Data processing occurs when the input is received, in opposition to the batch mechanism. A middleware receives the input and transfers it to the requested transaction, which can be seen as a set of programs. Examples of transaction processing systems include IMS (Information Management System) and CICS (Customer Information Control System). IMS traditionally uses a hierarchical database (DL/1) whereas CICS supports relational database (DB2), DL/1, and files.

Mainframe platforms also have daemon equivalent programs called Started Task (STC). An STC is a basic unit of work that usually runs as a background process, without a specific ending. They are initiated at a specific time, based on a command launch, a logon procedure, or another program.

2.1.3 Supervising elements

Mainframes' original design was highly constrained by the limitation of computing power. The platform has a detailed logging mechanism and powerful supervising tools as an Hypervisor.

Each LPAR, or sysplex of LPARs, contains its event logging mechanism. An event can be produced by the system or any product of the platform. They are identified by a hexadecimal type which permits to link the event to its producer: some type of range is reserved to IBM, between 00 and 7D, and some are not. At the exception of the type, most of the record fields are dependent on the natural events and will have different length and type -binary, string, others-.

The System Management Facilities (SMF) receives all the LPAR events, format and write them on the logging files[06]. To refer to a specific type of events logged by SMF, we refer to the hexadecimal type converted to decimal: the security product RACF writes record type 50 which is then known by SMF as SMF80.

Each unit of work on the platform has a common address space work. Events based on this address are called SMF30 and contain information about jobs (i.e. batch), STC (i.e. daemons), TSO sessions, OMVS (i.e. Unix Interface) running on z/OS.

With a superior abstraction degree, we can also retrieve information about the LPAR consumption based on the Hardware Management Console (HMC) HMC allows operators to see the current consumption of LPARs and propose to change different settings, such as the power management. An internal component known as the Support Element exposes the consumption of the monitored server on the HMC. Its operation is unclear and may or may not be based on an internal Wattmeter.

2.2 Hardware and hybrid consumption evaluation approaches

When seen as a physical computer, resource consumption metrics of a Mainframe can be obtained using hardware equipment. We briefly review the current state of the art of hardware consumption evaluation, and then argue about its mainframe declination which must be completed with software elements -leading to a hybrid approach-. Finally, we precise the limits of this approach.

2.2.1 State of the art in consumption evaluation: hardware and hybrid approaches

Hardware method consumption evaluation uses a hardware device to measure the energy use of a component, as a whole. It is the most accurate approach to obtain the instantaneous power consumption of physical components, expressed in watts. The perimeter can be lower than the computer if a power meter is used on internal hardware components.

This method can be used on a "bottom-up" approach. Using external power meters on the different components of the architecture, we can measure the consumption of a specific action -database operation, business treatment, or others- conditionally upon its isolation[07][08]. Based on the observation of the dynamic power consumption, software behavior could then be adapted to limit the use of one or multiple hardware components. When consuming components are clearly identified, another approach consists of building internal power sensors. High-performance servers can use it[09], especially on the CPU consumption.

Hybrid methodologies combine hardware components and a software evaluation to improve accuracy, or to limit measurement noises in a not-isolated environment. Because of the software aspect, the computer is self-aware of its own consumption and can adapt its behavior[10].

Hardware components can also implement and expose hardware performance counters (HwPC). The Running Average Power Limit (RAPL) is a major enhancement on CPU consumption evaluation. It provides accurate energy readings for all Intel processors superior to "Sandy bridge" micro-architecture (2011) and superior to "Zen" for AMD (2017).[11].

2.2.2 Mainframe declination

Using an external power meter, the ideal experience involves a non-used mainframe dedicated to our measurements. The platform could be used without any LPAR installed, to measure the static consumption of the PR/SM hypervisor and a single LPAR configuration could provide accurate metrics on the static consumption of z/OS.

Measuring more precisely the consumption of mainframe components has an uneven difficulty. Storage operations on disk and tape drive can be performed on dedicated storage equipment linked to the mainframe, which facilitates the measurement of these specifics components using another external power meter. However, measuring the consumption of CPUs, cryptographic coprocessors, and other internal components is complex. Most mainframe parts are non-standard, do not contain the RAPL interface, and would need complex electronics adaptation.

For practical reasons, this study could not be performed on a dedicated mainframe. Therefore, measuring the consumption of a platform includes the operation of multiple LPARs. If they can't be physically isolated, however, a hybrid consumption evaluation method could be applied.

IBM measures the amount of processing work an LPAR can perform using million service units (MSU) and million instuctions per seconds (MIPS). MSU is an artificial metric used for software cost that could be used to estimate the charge of a current partition. Knowing the total MSU consumed by all the LPARs hosted on the mainframe during a physical power measurement would permit to attribute power value to an MSU and a watt consumption to an LPAR based on its cumulated MSU. MIPS could also be used and should be more accurate because the PR/SM consumption is included.

2.2.3 Limits

Mainframe platforms are highly customizable. IBM z15 can have between 512 gigabytes and 40 terabytes of memory, up to 190 processor units, can be cooled by air or by water, and can contain dedicated coprocessors to cryptographic functions[12].

Software configuration also has a primary role by defining LPARs architecture, OS installation, or processor specification. Central Processors (CP) can be characterized to become specialized as System z Integrated Information Processor (zIIP) to support specific DB2 database work, System z Application Assist Processor (zAAP) to compute Java, and XML workloads, Integrated Facility for Linux (IFL), Integrated Coupling Facility (ICF) to manage Sysplex and others.

These configurations probably lead to heterogeneous performance based on the current types of workload which should be properly tested using a dedicated mainframe. However, this approach needs materials and human resources which can't be obtained as part of this study.

This approach is also inaccurate to measure the resource consumption of a specific workload of small sizes, compared to the mainframe configuration, due to the static measurement precision of LPAR installed operating system, STC and if MSU are used, PR/SM hypervisor.

2.3 Software consumption evaluation approach

The software approach estimates the resource consumption of a program based on metrics available in the operating system. This method is easier to install compared to the previous hardware involvements, even if the approach can be dependent on hardware facilities. It can be used in a non-dedicated environment which is crucial due to mainframe platforms being operational 24/7. This approach is also privileged because of its simplicity, which led to easier replication.

2.3.1 State of the art of software consumption evaluation approach

The software consumption evaluation is mostly based on a single component approach. Memory, network, and disk do have a power consumption which is usually not considered to simplify the model. In most cases, only the CPU consumption is considered. It has two advantages: its ease to be measured and its involvement in most of the hardware activities. However, this approach is not as accurate as the hybrid or hardware ones.[13]

A CPU consumption is usually based on process time. The amount of time used by a unit to process instructions of a program is measured in seconds or clock ticks. It does not include idle time and I/O time which makes it different from the elapsed time. If the CPU capacity is known, it can be expressed as a percentage called CPU usage. In a multi-processor environment, total CPU time is obtained with the sum of each process time.

The energy consumption granularity depends on the used tool. The original software approach was limited to platform consumption, but more detailed figures can be obtained due to its access to running processes. Software-based power monitoring can estimate the consumption of various software components, such as Greenspector for mobile devices[14], SmartWatts for containers, and others[15]. The detail level of this type of solution orientates software optimization with better architecture choices and code performance improvement.

To evaluate the global consumptions, an analysis of metrics (memory, disk, network) computed by the kernel is possible. However, results tend to be inaccurate[11]. Power models used by software-based approach can be divided into two categories: models based on performance counter and models based on resource utilization.[16].

The use of a performance counter is mainly based on HwPC from the RAPL interface of the processor. HwPC are highly correlated with processor power consumption. These metrics are used to train precise power models. However, the training phase is at least partially platform-dependent which leads to a design dynamic power model. SmartWatts is an example of a dynamic power model[15].

Resource utilization refers to virtual or hardware components. This type of model treats components activity as an input and converts it to estimated power as output. Virtualization can be considered by monitoring the use of virtual components such as virtual CPU (vCPU). Resource utilization models can be accurate, but they cannot link the consumption to internal processes[16].

2.3.2 Mainframe declination

SMF30 is the record type written by the Common address space work. It is mainly used for accounting purposes; however, it contains accurate metrics about a unit of work in the LPAR. A record is generated at a work unit start, during its activity, and at its end completion. It contains global fields to identify the type of work and different sections related to the work activity.[06]

The Processor Accounting section contains different CPU times related to the address space activity. Each time is expressed in hundredths of a second (1/100 second). Reported process time includes the global time (SMF30CPT), CPU time used by the Service Request Block (SRB, system tasks used on behalf of the program, SMF30ISB), CPU time used by the Task Control Block (TCB, time used by program statements, SMF30ICU), amount of CPU time used to process I/O interrupt (SMF30IIP), and others.

The I/O activity section describes reading and writing operations performed on direct-access storage devices (DASD) such as disk drives. Metrics include DASD I/O connect and disconnect time (SMF30AIC and SMF30AID, respectively), and total blocks transferred (SMF30TEP and SMF30TEF). However, dataset block size should be considered on these latest values. These fields can be completed by the storage section which contains different statistics about the kind of storage used by the work unit.

These metrics could be used instead of the HwPC and resource utilization ones, to train an efficient power consumption model. The training phase needs representative power consumption of the LPAR, which can be obtained based on the activity of the partition described as previously seen in MIPS, the activity of the platform, and the consumption of the platform.

The PR/SM hypervisor allows an operator to visualize the current consumption of a mainframe server based on the previously described HMC. However, no other granularity can be seen, the consumption of a specific LPAR is not exposed and is difficult to isolate in the case of a production environment hosted as a plex on multiple separated mainframes.

2.3.3 Limits

PR/SM does propose an API to access some metrics using HMC. However, we weren't able to obtain access to these functionalities due to security reasons. The MSU and MIPS consumption of an LPAR is retrievable from the LPAR itself, but the lack of possible comparison with the MSU/MIPS and power consumption of the platform limits the real-time training of a model. Moreover, PR/SM metrics are not freely available because the hypervisor manages multiple LPARs, some being sensible for the group activity.

In consequence, model training must be performed on static data. A model is determined on previous observations and then applied to the studied LPAR for our analysis. The use of MSU to estimate the workload of a partition has limits. MSUs are an artificial metric, which tends to give different values at each new mainframe computer: IBM lowers the value, which decreases software invoices based on it and encourages customers to buy the new model. MSUs are platform-dependent which doesn't allow the model to be reusable. MIPS however are constant and already used to compare different Mainframe performance, its use should be privileged.

Metrics available in z/OS are easier to obtain and manipulate because a dedicated LPAR can be used. Once a model is established, it should be the only one involved. A correlation must be established between the MIPS consumption of a job and more standards metrics. The performance section of SMF30 allows this to be done by giving the service unit (SU) consumption at the work unit level. SU is also detailed as CPU service units, SRB service units, I/O service units, and main storage occupancy (MSO) service units.

Chapter 3

Mainframe consumption estimation

To the best of our knowledge, there are no mainframe consumption studies in the current scientific literature. We propose to humbly contribute by providing leads on how to estimate its consumption, based on its characteristics, and how to compare it to more standards platforms. This work seems important to understand the impact of the traditional banking system and could allow a better architecture design. It is part of the manifesto for energy-aware software[03] to be able to have accurate consumption metrics to understand the current trends which permit to choose solutions, at all granularities, based on these and other performance metrics. First, we study in 3.1 the z15 server consumption. Then, we extend the study to mainframe platform consumption in 3.2, and finally we estimate the mainframe workload in 3.3 and compare it to a x86 equivalent.

3.1 Mainframe z15 servers

Mainframes are physical servers that most of the time work in a cluster for resilience reasons. They host multiple LPARs which use various types of processors exposed by the hypervisor. In this part, we will study global consumption aspects based on the example of Groupe BPCE mainframe servers.

3.1.1 Mainframe servers processors utilization

The mainframe server configuration observed is as follows: 18 central processors, eight active ICF processors and eight inactive ICF processors, 24 SAP processors,

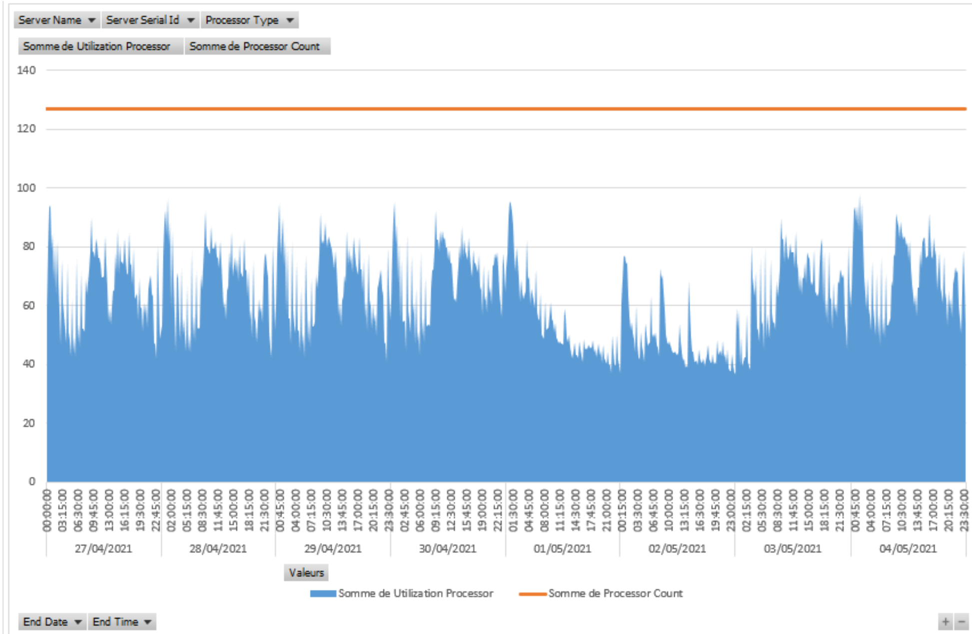

Figure 3.1 – Utilization of LPAR visible processors of mainframe servers

six ZIP processors, and one IFL processor.

In the five types of processors listed in this configuration, the inactive ICF processors, the SAP processors, the IFL processor, and the other 43 plugged-in but inactive processors can only be accessed by the HMC view because they do not directly serve the partitions. From an LPAR, only the central processors which manage most of their work, the active ICF processors, and the ZIIP processors can be seen and compared to a workload.

For these three types, a history of processor utilization can be retrieved in the SMF, as they are present in LPAR log files. For the other type of processors, the SMF cannot be used and only the current utilization can be seen from the HMC. There are no historic functionalities and without access to the HMC API, the monitoring possibilities are limited.

In figure 3.2, the utilization of each LPAR visible processor in the mainframe cluster is aggregated. It has been obtained by retrieving the number of processors for each type and by multiplying it by the associated percentage of utilization of

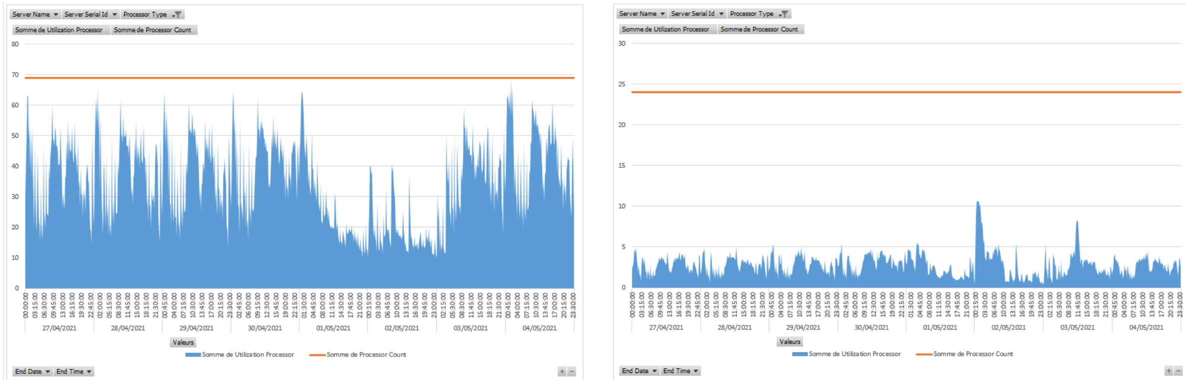

Figure 3.2 – Utilization of CP (left) and ZIIP processors (right) of mainframe servers

the type. For reference, the total number of LPAR visible processors is also shown. During our week of observations, the average utilization used 63.4 CPUs, about 50% of the 127 total processors, with a standard deviation of 14.6 CPUs. This deviation is much more significant than the observed deviation of consumption, 22% of the average against 1.3%, and tends to confirm that partition workload is not correlated with consumption. We can also observe that the total number of online processors is approximately equal to twice the average. The loss of a mainframe would reduce the number of total processors to 95, which would be enough to support the workload 99.5% of the time, with a percentile of 99.5% at 94.8 processors, with no need for the backup processors if we abstract the types.

We then study more in detail the utilization of the different types of processors. A global approach is not enough due to the heterogeneous workload and configuration of each processor type. In figure 3.3, the utilization of central processors and ZIIP processors is described. On average 34.7 central processors are used with a standard deviation of 13.8 processors. The loss of a mainframe would reduce the 67 central processors to a possible 49 because the current configuration is between 18 and 16 central processors per mainframe. The need for central backup processors is then justified because that configuration will not be able to support the same workload 18% of the time, with a percentile of 82% at 48.7 processors. To be able to process all the workload experienced during this week with a constant number of central processors, a minimum of 69 operational units is necessary, which corresponds exactly to the current allocation.

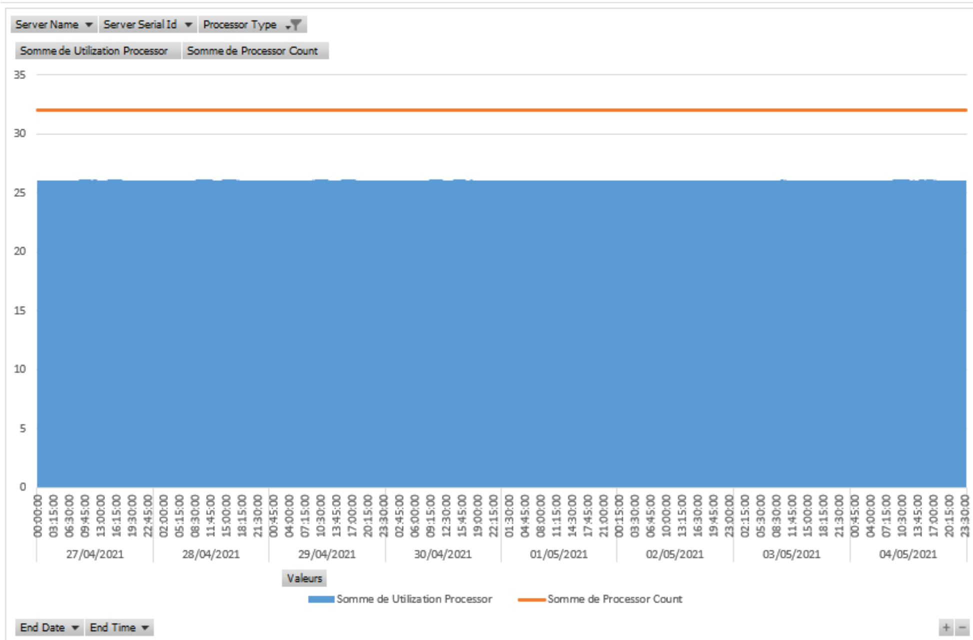

Figure 3.3 – Utilization of ICF processors of the mainframe servers

However, this allocation differs for ZIIP processors. They are in charge of specific workloads, especially related to database operation. With an average CPU use of 2.8 units and a standard deviation of 1.4 units, they never exceeded 10.7 units on their rare pics. We can then question the configuration of 24 ZIIP processors. It seems to be highly exaggerated by covering 118% of the highest workload seen during the week.

Finally, the last processor type visible from the LPAR is the ICF processor. As a coupling facility, they are constantly fetching data to reduce coupling time, which explains the very stable utilization seen in figure 3.4. During a week, they used an average of 26.1 processors with a standard deviation of 0.01 units. Due to their behavior, we can also question the necessity to have 32 processors, even if this allocation appears less excessive than the ZIIP one.

3.1.2 Consumption of mainframe servers

The utilization of each processor type is exposed by HMC on all mainframes. The BPCE mainframe cluster is composed of four z15, called s1 to s4. Each z15 contains

108 processors plugged in. There are currently 65 processors allocated, whereas 43 are not in use and serve as backup for a workload increase or any technical failure of the other processors. Even if they can be seen as offline, these backup processors are powered on, which means that their use will not drastically change the current power consumption of the central processor complex (CPC). It is described as a way to reduce the time necessary to change the processor allocation operations and therefore allow better recovery time after a hardware failure. However, it implies a static consumption of 43 processors that most of the time are not required. High availability requirements lead to an architecture where a mainframe must be able to retrieve all the workload of another mainframe server, leading to oversized design.

Mainframe processors have another particularity, they operate at a fixed clock frequency of 5.2Ghz on the z15. Modern standard PCs and servers have a flexible clock rate between a minimum and a maximum CPU frequency. The base CPU frequency mentioned on all CPU models lies between these two values. This frequency is then adapted based on the current workload which can lead to variation in the consumption of other components such as the cooling system, the DRAM, the hard disk, etc. Mainframe processors operate at a fixed rate which tends to improve the response time of a workload pic.

In figure 3.4 we extracted the consumption of one of the four mainframes in the cluster during a standard week from the HMC. The mainframe consumption was noted every 15min. We observed an average consumption of 9.58 kW with a standard deviation of 0.13 kW. The low standard deviation was expected as the CPU consumption is mostly based on the CPU frequency. The previously described CPU behavior leads to a consumption independent of the workload experienced by the server because of the fixed number of CPUs, and the fixed frequency of each.

Each mainframe of the four-cluster has between 31 and 32 processors, leading to very similar consumption for s2, s3, and s4. Therefore, the total consumption of the four mainframe servers can be estimated to be a static 38.32 kW.

3.1.3 Limits

With real data extracted from the BPCE mainframe environment, this part aimed to show the consumption behavior of a mainframe server. The variation of the workload managed by the LPAR is mainly observed with the utilization of the central processor with different waves between the night batch process and the day basic

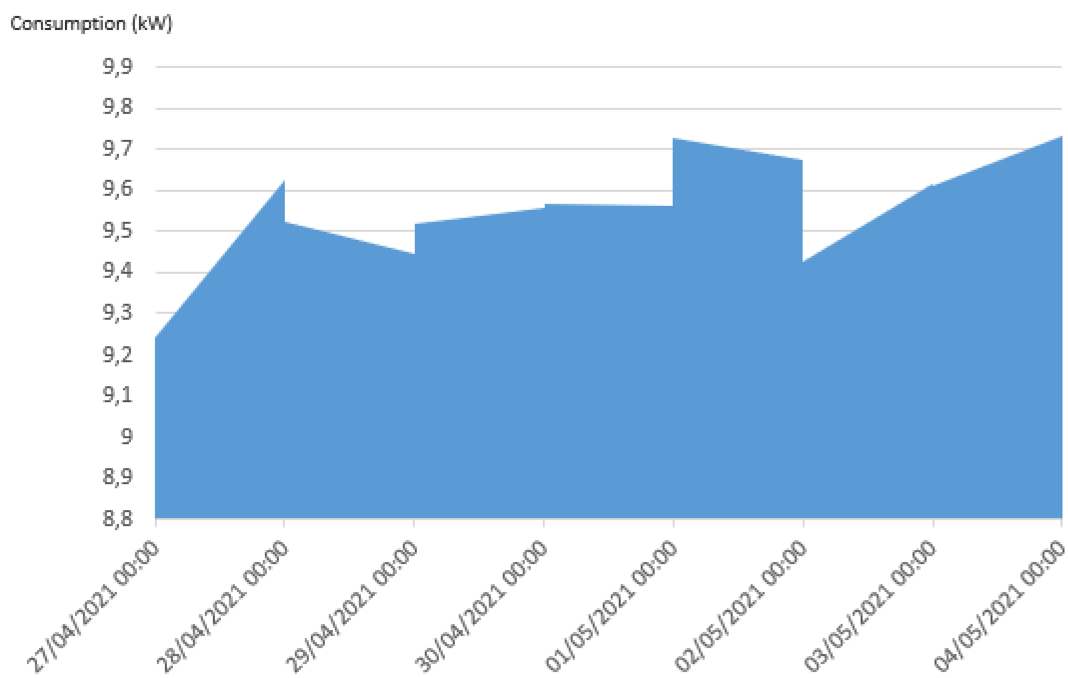

Figure 3.4 – Mainframe server consumption evolution

activities. However, this workload does not impact the server's global consumption.

Despite being the most present processor type, central processors represent only 69 units on a total of 260 active units. Moreover, the consumption of the 69 units should be compared to the sum of active and inactive processors, because all processors are plugged in and powered up in a mainframe server. Therefore, the central processors do represent only 16% of the total number of processors, 432, in the 4 servers.

The second reason is based on the behavior of the mainframe processors. The fixed frequency tends to decorrelate consumption and workload. The traditional software approach to estimate the power consumption of a server, based on CPU activities and performance counters is then inadequate for mainframe servers.

3.2 Mainframe global architecture

As we were not able to find any previous work related to mainframe consumption evaluation, choices were made to focus on the mainframe server itself as a first step. However, the mainframe platform used dedicated external material: storage servers, virtual or physical tapes, and network elements. The state of art software consumption platform is mostly based on CPU utilization which in a mainframe ecosystem corresponds to the mainframe server itself, but other components may have more impact than they usually have in the distributed world. We now propose to study their consumption and deduct an overall carbon footprint.

3.2.1 Estimation of global consumption

To estimate the global consumption of the mainframe platform, we must add to the four-mainframe server's consumption obtained in 3.1.2 the external materials, but also the two backup mainframes which we have not described so far.

The two backup mainframes servers are available in a distant location as a part of the recovery plan. They are synchronized with the production mainframes and can take over the workload if the four mainframe servers are unavailable. They are two z14 servers which is the previous mainframe generation. We assume their consumption to be similar to the z15 one and therefore the global mainframe server's consumption to be 57.48 kW.

The Groupe BPCE mainframe platform is currently composed of four z15 servers, two z14 servers, six IBM DS8950F, four IBM FICON SAN512B-6 Director, four IBM TS7770, two IBM TS7770T, and one IBM TS4500. Below are the details of each of the components non-already explained. For our estimation computation, we must point out that we did not have at our disposal monitoring tools as detailed as the ones used for the z15 consumption. It may exist dedicated HMC for each component, but their existence is not publicly known in the company.

An IBM DS8950F is a flash storage array composed of up to 20 processors and which can contain up to 5.9 PB of flash Tier 0, Tier 1, or Tier 2 storage. According to internal documentation, an IBM DS8950F has a peak electric power of 9.8 kW. As it is a peak and not actual consumption, we refer to the difference of 79.3% between the announced peak power of a mainframe and its real consumption to report the same ratio. It may be a strong hypothesis as the studied material does not have the same purpose but we didn't have real metrics at our disposable. With this method, we estimate the total consumption of the six storage servers to be 46.6 kW.

Both IBM TS7770 and IBM TS7770T are virtual tape libraries. They contain hard drives that emulate tape storage for the mainframe operating system. Due to legislation and recovery needs, there is also a non-virtual tape library: the TS4500 which archives data on physical tapes. They respectively have a peak consumption of 6.96 kW, 3.48 kW, and 12.72 kW which corresponds with the previous estimation method to a total consumption of 37.7 kW for the 7-tape equipment in use.

Finally, we have the four FICON elements which act as a motherboard by connecting the mainframe servers to its flash storage arrays and their tape solutions using fiber optics. A unit has a peak consumption of 2.59 kW which corresponds to a global consumption of 8.21 kW according to our estimation method.

As we deducted global consumption from a static ratio of the peak consumption, the global consumption may not be accurate. However, it is enough to have an order of magnitude of the real consumption. It should be noted that the consumption is subject to evolve as three IBM DS8950F were recently added to the architecture, which doubles the flash storage consumption. The global consumption of the Groupe BPCE mainframe platform is estimated at 150 kW.

Figure 3.5 illustrates the distribution by equipment of the global platform consumption. Previously studied servers represent only 26% of the global consumption,

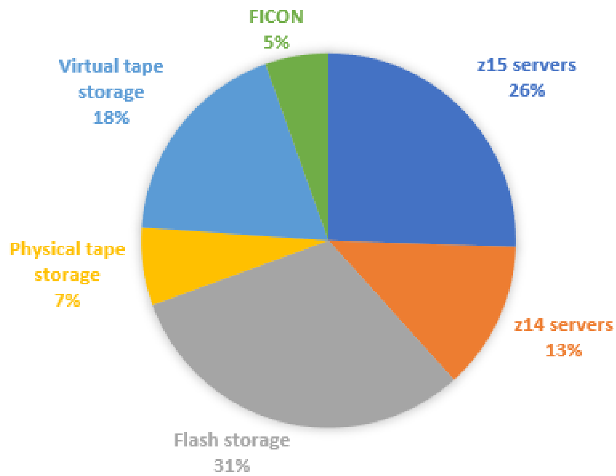

Figure 3.5 – Groupe BPCE mainframe platform consumption per equipment in 2021

which justifies that mainframe consumption studies cannot be limited to the CPU utilization. State of art consumption evaluation methods are therefore inadequate for the mainframe ecosystem. Flash storage solutions are the most consuming category by representing 31% of the global platform consumption. If we aggregate flash storage and tape storage solutions, 56% of the global consumption is dedicated to storage equipment.

3.2.2 Annual carbon footprint

We established the global consumption of the platform. This is a necessary step to measure the impact on the environment that is usually estimated with the carbon footprint[17]. Based on a projection of the annual consumption and the manufacturing process, we now want to study in more detail the platform's CO2 emission.

The Groupe BPCE's mainframe platform is located in the south of France. Based on Électricité de France (EDF) metrics, an average of 36g eq. CO2 was produced per kWh in the last 15 years. As previously established, the platform consumes 150 kW which is equivalent to 5.4 kg eq. CO2 per hour or 47.3 tons eq. CO2 per year. This emission highly depends on the electricity production methods of the hosting country. According to [19], emission per unit of power generation is 522 g/kWh in the United States and 721 g/kWh in China. An equivalent mainframe platform in these two countries would respectively produce 685 tons and 947 tons of eq. CO2 per year.

Category	Equipment	Manufacturing carbon footprint (kg eq. CO2)	Number	Lifetime (year)	Total (kg eq. CO2) / year
z Server	z15/z14	58 560	6	3	117120
Flash storage	IBM DS8950F	48 864	6	5	58636,8
Virtual Tape solution	IBM TS7770	47 744	4	5	38195,2
	IBM TS7770T	23 872	2	5	9548,8
Physical Tape solution	IBM TS4500	56 768	1	8	7096
FICON	IBM FICON SAN512B-6	4 672	4	5	3737,6

Figure 3.6 – Manufacturing carbon footprint per equipment

Based on DELL's estimation, a traditional computer produces 300 kg eq. CO_2 during its manufacturing process. The Green IT association in a partnership with Fujitsu estimates it to 339 kg eq. CO_2. It includes the extraction of resources, assembly, and transportation. We will take the average estimation of 320 kg eq. CO_2 to estimate the cost of the mainframe manufacturing process. We assume that a computer's weight is correlated to its carbon footprint and that a traditional computer tower weighs 10 kg. The total weight of the Groupe BPCE production mainframe platform, composed of four z15 and two z14, is about 10,980 kg which can be roughly estimated to have a carbon footprint of 351 tons eq. CO_2. Based on the mainframe server's lifetime, detailed in part 4.3.1, the annual carbon footprint due to the manufacturing process is about 117 tons eq. CO_2.

To the best of our knowledge, there is no carbon footprint valuation for FICON equipment, tapes, and flash storage solutions. We believe that a more appropriate way to estimate their carbon footprint is to assimilate them to servers and, therefore, use the same method previously described. Based on internal documentation, a FICON equipment weighs 146 kg, an IBM DS8950F 1,527 kg, an IBM TS7770 1,492 kg, an IBM TS7770T 746 kg, and an IBM TS4500 1,774 kg. Details of manufacturing carbon footprint for each equipment are summarized in figure 3.6. The useful life of each equipment may vary as it is based in part on technological changes.

The annual global carbon footprint can now be estimated by the addition of the manufacturing carbon footprint per year and the annual consumption. At this stage, we also consider the Power Usage Effectiveness (PUE) of the Groupe BPCE data centers, i.e., how much energy is used by the data center for each watt consumed by the mainframe platform. The platform is dispatched between two different sites, approximately 2/3 of the equipment is at a data center with a 1.58 PUE and 1/3 at a data center with a 1.71 PUE. We choose to use an average of 1.62 PUE to simplify.

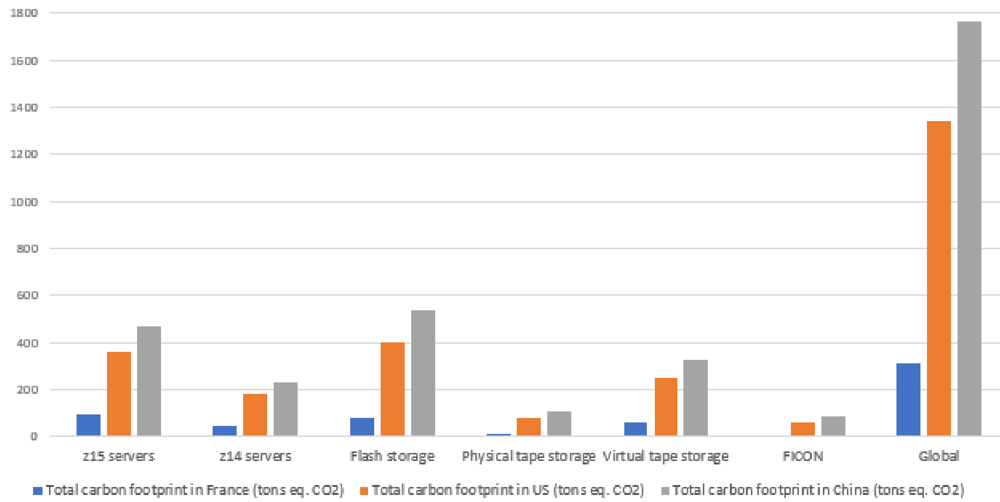

Figure 3.7 – Global carbon footprint per equipment on three countries

In France, the manufacturing phase represents 75% of the carbon footprint of the studied platform due to a relatively low carbon energetic mix on a total of 311 tons eq. CO2. However, in the United States, the same platform would have a carbon footprint of 1,345 tons eq. CO2 with 83% of which is due to electricity consumption. In China, the carbon footprint would be about 1,769 tons eq. CO2 with 87% due to electricity consumption. Details on carbon footprint per equipment on the three countries can be found in figure 3.7.

This illustrates the importance of the host country's energy mix and the need to develop low-carbon electricity production sites in both China and the United States. Low carbon energy production includes Hydro and Wind energy, Solar PV, Biofuels, Geothermal, Nuclear fusion and fission[20]. According to EDF, France electricity is mostly produced by Nuclear power, 69.9% in 2019. Others countries manage to have a higher hydroelectricity part based on their number of available natural sites, such as Switzerland, Norway and Island. However, in the case of the United States and China, the energy production is mostly based on thermal station powered by coal, oil, natural gas and wood. Based on the Internation Energy Agency (IEA) data of 2019, fossil fuels represent 62.5% of electricity production in the United States and 69.9% in China.

We may note that some companies that own data centers in these countries, such as Google and Amazon, compensate their electricity consumption with purchases of renewable energy. As electricity cannot be distinguished based on its source once in the power grid, it is unclear how efficient is this method but it may support the creation of new low-carbon energy production site. Improving the PUE of data centers is also an efficient way to reduce the carbon footprint, especially when electricity consumption is the main cause of carbon footprint.

The manufacturing footprint is also dynamic and could be reduced by increasing the lifetime of the equipment. As it is the main carbon footprint cause in France, it may be an area for improvement, especially on z15 and z14 servers which represent the largest part.

3.2.3 Limits

Despite the approximation induced by our estimation method on the consumption of non-server equipment, we should note that the estimated carbon footprint may be underestimated.

First, indirect data center materials used by the mainframe platform are not included in our computation. For example, backup electricity generators equip each data center and are sized according to data center consumption, which is impacted by the mainframe platform consumption. However, in 2019, the global data center consumption of Groupe BPCE was about 70,290,065 kWh. The 243 kW mainframe consumption, obtained by the previously estimated 150 kW multiplied by the 1.62 PUE, corresponds to 2,128,680 kWh in a year. 3% of the data center consumption is therefore attributed to the mainframe platform, with tends to indicate that the platform has a negligible impact on data centers.

We also did not consider network communications. EcoInfo, a CNRS grouping of French research teams, estimated the emission of 1 Go transmission on the RENATER network, a network dedicated to French universities. Between two sites 700 km apart, the carbon footprint was estimated to be 1.4 g eq. CO2. Groupe BPCE has its own duplicated fiber-optic network which may have a different consumption, but we assume that the consumption can be estimated as twice that of the RENATER one. Figure 3.8 represents the annual network activity of Natixis, one of the companies of Groupe BPCE. Natixis' 11 dedicated LPARs managed a total of 257.37 Go between October 2019 and September 2020. We assume that Natixis' activity is representative of the other 69 LPARs and therefore estimate that the mainframe platform manages 1,614 To, leading to a carbon footprint of 4.5 tons eq. CO2. This

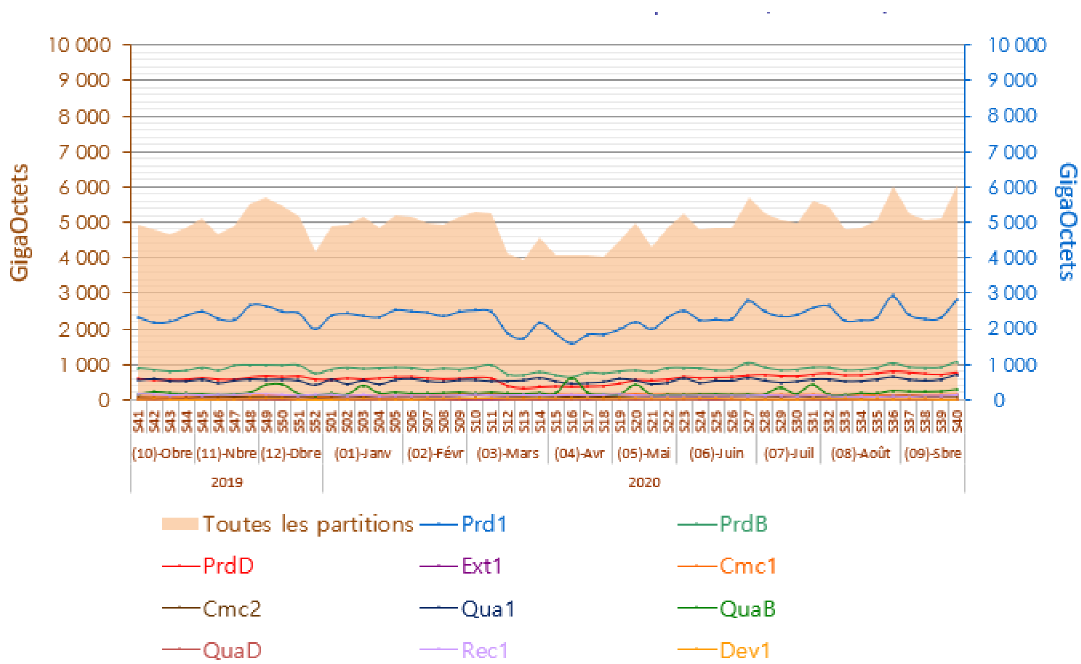

Figure 3.8 – Natixis mainframe network activity between October 2019 and September 2020

number was not added to facilitate comparison with non-mainframe platforms, as we assume the network activity will not change based on the architecture. This is however a simplification, as micro-architecture are known for their increase of network activities[21].

Finally, we choose to limit our carbon footprint study to the technical elements. A more global approach would include the carbon footprint of developers, PCs, and all related infrastructure[17]. We tried to be exhaustive on the direct components, but the indirect ones, for example, servers related to the mainframe, are harder to estimate. Regarding the employees, it is difficult to find the total number of mainframe jobs due to the Groupe BPCE's structure being composed of different companies. However, we add some elements of analysis in part 4.1.2.

3.3 Estimation of mainframe workload for comparison needs

We now have an order of magnitude of the mainframe platform consumption and its carbon footprint. Now, we would like to bring some comparison elements to estimate what the mainframe workload could consume in a distributed environment. In 3.3.1, we study how previously described mainframe metrics can be used on Groupe BPCE's platform. Then, in 3.3.2, we estimate an equivalent x86 architecture able to manage the workload. Finally, 3.3.3 compares both architectures.

3.3.1 Mainframe workload metrics

Due to the limits of MSU, being non-standard, inconsistent between generations, and not including PR/SM hypervisor consumption, we privilege the use of MIPS. To the best of our knowledge, MIPS are not retrievable by a program in the LPAR, which could justify the use of MSU for a dynamic training model. For our study, a static evaluation is however enough, especially with the previously CPU static consumption described.

The internal documentation indicates the available MIPS central processors. It varies slightly for each z15 due to the number of central processors per server: the two 17 CPs configuration have 26,032 MIPS announced, the 16 CP configuration has 24,850 MIPS and the 18 ones have 27,213 MIPS. We can observe that MIPS are not completely proportional to CP numbers. As the rest of the processor's configuration

LPAR Configuration				Full CPC Capacity (based on usable RCP count)					
	Identity	Hardware	SMT	GP*	zAAP	zIIP	IFL	ICF	Total
#5	BPCE-IT - B1 2021T2 v1	8561-T01(Max108)/700: GP=17 zIIP=6 IFL=1 ICF=8	✓	25 892	n/s	12 635		11 861	50 387
#6	BPCE-IT - B2 2021T2 v1	8561-T01(Max108)/700: GP=18 zIIP=6 IFL=1 ICF=8	✓	28 342	n/s	12 270		11 851	52 463
#7	BPCE-IT - B3 2021T2 v1	8561-T01(Max108)/700: GP=17 zIIP=6 IFL=1 ICF=8	✓	25 125	n/s	12 217	2 038	11 817	51 197
#8	BPCE-IT - B4 2021T2 v1	8561-T01(Max108)/700: GP=16 zIIP=6 IFL=1 ICF=8	✓	25 209	n/s	12 527	2 040	11 809	51 585

Figure 3.9 – MIPS capacity of Groupe BPCE mainframe platform estimated by zPCR

is identical, it may be linked to the PR/SM resource reservation behavior and the loss of efficiency due to parallelism synchronization.

As CPs represent only one of the types of processors, we used Processor Capacity Reference for IBM Z (zPCR) software to assume the global MIPS capacity of the platform. zPCR is designed to provide capacity planning insights for mainframe platform processors running LPAR configurations. With the current configuration, where we only wrote the allocated processors and not the backup ones, results are visible in figure 3.9. The MIPS CP estimation is slightly lower, by 0.58%, than the one written in the documentation, we chose to use the estimation as it is based on the current configuration. The total mainframe MIPS capacity is therefore 205,632 MIPS.

3.3.2 Equivalent x86 architecture for the mainframe workload

We now want to establish what an equivalent x86 architecture would be to manage the estimated workload of a mainframe. As the mainframe consumption is mostly static, we know that the current platform would have a similar consumption with variable MIPS values. In part 3.1 we established that the CP configuration corresponds to the actual use, without significant oversize, and decided to use the maximum MIPS capacity as the referential value.

Even though instructions per second is a common metric, we observe a difference between a mainframe and x86 instructions. The mainframe platform MIPS amount is indeed lower than a single modern Intel Core i7 processor such as the i7-8086K with 221,720 MIPS. This is an incoherence as the six cores at 5 Ghz of the Intel processor cannot overpass the MIPS of 68 processors with 12 cores at 5.2 Ghz, even with differences in processor architecture. We should also note that the 68 CP are just a part of the platform processors. Therefore, we cannot assume that a mainframe

Company	Test	MIPS Equivalent	Transactional server CPU utilization (%)	Equivalent cores	Database server CPU utilization (%)	Equivalent cores	MIPS per core
Tmaxsoft	CICS workload (1672 TPS)	13676	79	12,64	92	14,72	499,8538012
	CICS workload (2600 TPS)	21268	76	24,32	84	26,88	415,390625
	Batch and CICS (1409 TPS)	11525	68	10,88	72	11,52	514,5089286
HP	CICS - 2CPU BL460	1489	99	1,98	N/A	N/A	752,020202
	CICS - 12CPU BL460	5738	76	9,12	N/A	N/A	629,1666667
	CICS - 80CPU DL 980 (+DB)	6678	41	32,8	N/A	N/A	203,597561
	CICS - 120CPU DL980 w/SQL on DL580	7671	29	23,2	16	6,4	259,1554054

Figure 3.10 – Resume of HP and TmaxSoft sizing tests based on mainframe MIPS

instruction is equivalent to an x86 one. It could be caused by assembler compiler differences or by different counting methods.

To determine an equivalent x86 architecture, we based our work on documentation proposed by HP and TmaxSoft to migrate mainframe workload on x86 servers[22][23]. They both use zRef, a z/OS batch and CICS application written in COBOL and using DB2 to simulate the activities of a financial trading firm. It claims to be designated as representative as possible of the actual mainframe application.

Figure 3.9 recapitulates the equivalent number of CPUs used per MIPS on experiences led by the two companies. We can see that only small platforms were tested, with a highly variable ratio between MIPS and the total number of processors effectively used. We spoke with an AWS Cloud architect responsible for a mainframe migration project who estimated, based on his own experience, that the number of cores needed was one per 150 to 200 mainframe MIPS. We were not able to confirm this ratio with other architects, but it tends to correspond to the most pessimistic experience of our benchmark, the "CICS - 80CPU DL 980 (+DB)" led by HP, where a DL980 server is used. It contains eight Xeon E7-4870 2.4 Ghz processors with ten cores each and is used to host the application and the data base.

We take 175 MIPS per core as our referential value, leading to the need for 1175 x86 cores to manage the current optimal capacity of the Groupe BPCE's mainframe platform. As the number of mainframe MIPS retained only takes into account the allocated processors, we must determine a more accurate vision, using the high availability criteria of a banking platform. It will be performed based on the number of equivalent servers in part 3.3.

3.3.3 Comparison and limits

An x86 platform is a generic term that can correspond to a multitude of architecture and server configurations. We cannot pretend to be exhaustive here, but we can establish an order of magnitude of what would be an equivalent x86 platform consumption and carbon footprint.

We take as a reference a Dell PowerEdge R640 server that is at our disposal in the EPISEN infrastructure. It uses an Intel Xeon Silver 4114 processor with ten cores at 2.20Ghz to run a Vsphere hypervisor. In a virtualized environment, each thread is represented as a vCPU in the instance leading here to the disposal of 40 vCPU. However, the given ratio of 175 MIPS per core does correspond to the physical core. Therefore, 118 servers alike are required to have a minimum of 1,175 cores. Compared to the 1524 mainframe cores, obtained with the 127 allocated processors with 12 cores, it represents a decrease of 23%.

However, the previously estimated mainframe platform consumption and footprint included the backup processors and recovery infrastructure. The platform environments do not all need high availability, and we can argue that only production and development, being in some cases the code deposit, require it. Indeed, the x86 architecture allows finer sizing. Based on Natixis network activity, figure 3.8, we established that production and development environments represent 81% of the activity. We assume that network activity is highly correlated to the MIPS used and that Natixis activity is representative of Groupe BPCE. In correlation, 81% of the servers, i.e., 96 servers, would need to be at least three times as many to be used in a backup and recovery infrastructure. This leads to an overall of 308 equivalent servers with an average CPU utilization of 38%.

We estimate the PowerEdge R640 consumption at 38% CPU utilization to be about 176 W by linear regression based on one-hour average server consumption at 90% and at 50% CPU utilization. Therefore, the overall x86 server's consumption can be estimated at 54.21 kW against 57.48 kW for mainframe servers, which represents a decrease of 5%. We used the previous carbon footprint with a server weight of 21.9 kg to estimate a manufacturing carbon footprint of 216 tons eq. CO2, which represents a decrease of 39% compared to the mainframe platform one.

The other equipment, flash storage servers, tape libraries, and network elements can be reported as is. We have no reason to believe that they would behave differently in an x86 architecture.

With an average life of five years for servers, the annual carbon footprint of the x86 equivalent architecture is about 235 tons eq. CO2 in France, a decrease of 25%

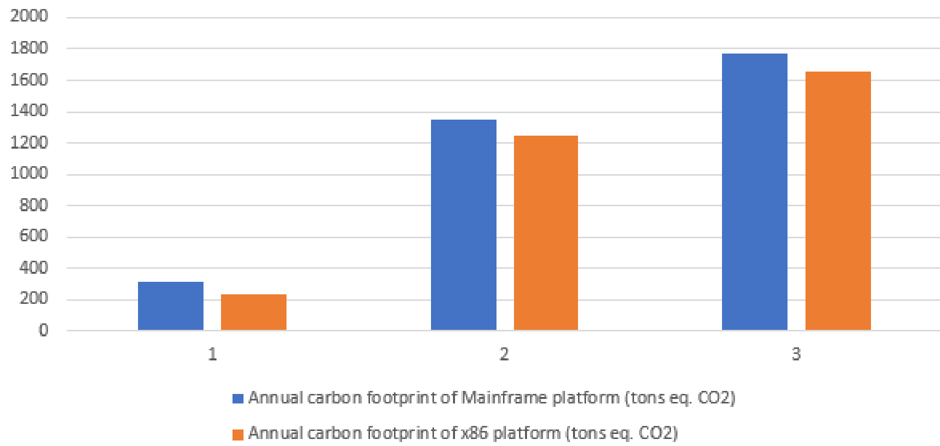

Figure 3.11 – Comparison of platforms global carbon footprint in France (1), in the United States (2) and in China (3)

mostly due to the manufacturing carbon footprint. Reductions are less significant in the United States, 7%, and in China, 6%, due to the impact of the energetic mix. Figure 3.11 provides some elements for comparison, but we must note that these gaps are not significant due to the approximations used in the sizing and carbon footprint estimation. For example, the use of a 150 MIPS per core ratio would have led to the necessity of 359 servers and therefore a slightly higher consumption of the x86 platform compared to the mainframe one of 4% and a closer carbon footprint. We can however say that a mainframe platform is likely to have a similar consumption to an x86 platform based on our observations.

The limitation of this study also comes from approximations made for the x86 architecture. To simplify our computation, we did not distinguish different types of servers by type of workload. The use of a multi-processor server may also be more appropriate for some workloads. We also took the hypothesis of portability as is, which is the scenario proposed by cloud providers, and not a rewrite of software elements. Furthermore, the mainframe workload used for our comparison may not be representative of all mainframe platforms. Finally, our study is partially based on commercial studies which may not correspond to scientific standards.

Chapter 4

Perspective of mainframe consumption

Having established the current state of consumption, it is now interesting to consider what could be the place of the mainframe ecosystem in the future. Beyond the usual rumors of disappearance, it seems that no previous work tries to address this question with QoS criteria, but also with sustainable IT elements. We will first study the specificities of mainframe software in part 4.1, then we will describe the mainframe ecosystem evolution in part 4.2, and finally, we will think about the place of the mainframe in Green IT, in a voluntarily simplified definition adapted to our context: Information and communication technologies (ICT) whose consumption has been voluntarily reduced.

4.1 Mainframe software elements

Mainframe software can be considered specific because of the languages used, Cobol, Fortran, DL1, REXX, but also because of the technologies behind it. This part aims to see how mainframe software and sustainable IT are related.

4.1.1 Optimization based on the state of art

Mainframe software can be classified into two categories. Most of the business code is written for transactional managers: this is our first category. Code written outside transactional managers covers different work, but we will only take into account batch jobs because others usually contain less business code and consume far less, as studied in 3.2.

As described in [24], batch jobs and interactive operations can reduce their brown energy consumption by being aware of the energy mix. It is a common practice to schedule the batch jobs at the most appropriate time[25]. To reduce the platform's carbon footprint, jobs can be dynamically scheduled at a time where renewable energy is available and has a significant share in the overall energy mix[26]. Due to its periodicity, this mix is not constant, leading to hours emitting more CO2 than others. However, mainframe jobs are only scheduled based on platform resources available and the same philosophy cannot be applied due to their static consumption over time.

However, the interactive operation cannot delay its response. The paper proposes an example of a controller that decreases the user experience when green energies produce less. The software can reduce its consumption by removing its optional answer functionalities. An example is given with a recommendation widget in a commercial shop. A similar mainframe controller could be implemented in some IMS and CICS to avoid some unnecessary database readings. Even if this improvement should be probably limited to the most used mainframe application, where data volumes would be enough to notice a reduction in consumption, the same limits induce static consumption of the applied platform.

Mainframe platforms cannot benefit from these types of upgrades. It seems that improvement of the current direct consumption can only come from usage reduction or hardware optimization. We should nevertheless mention that the current software and platform rental billing method is partially based on mainframe usage. Some advantages given to the "pay for what you use" model are probably similar, by encouraging developers to reduce CPU consumption to avoid bypassing the limit bought in the current package.

4.1.2 Effect of stability

One of the strengths of the platform is its retro compatibility capabilities. Each new version of z/OS is designed to facilitate the execution of previously written code. This doesn't mean that development is never required, especially when some memory registers are changed, but it is rare and software adaptations are usually minimal. It is commonly admitted that a mainframe platform allows for greater code longevity: some banking applications have been in service for more than 40 years, when an equivalent in a Windows/Unix environment is, to the best of our knowledge, far more difficult.

The carbon footprint of the development phase is sometimes neglected. Most studies focus on the execution phase, and how the software impacts the running consumption[17] but to be more representative, we should also consider the development and deployment phases. One way to measure this is to study the carbon emission of the different resources required for the development. They can be resumed as the number of employees required and their means of transportation, the number of direct equipment, such as the PCs, and the number of indirect equipment, such as the server allocated for the development.

Stability leads to fewer needs in maintenance development, which reduces the different resources used, and their footprint, in this phase[18]. Although metrics are hard to obtain on this specific topic, we can argue that a well-optimized code running for 40 years has a better footprint than a similarly optimized one with frequent evolution. The number of employees necessary for the development is reduced just like the number of devices required. For nine employees, the travel footprint can go up to 23 tons of CO_2 per year added to the 24 tons of CO_2 necessary for the associated IT equipment[17]. Banking groups employ far more mainframe developers than the example taken by the authors, leading to higher savings. Finally, we can also point out that mainframe development is made on the same mainframe platform, which doesn't need other infrastructure than the one used for production operation.

4.1.3 Architecture effects

Despite the already described advantage of the billing method, the mainframe platform also has other advantages that tend to reduce its consumption. We cannot pretend to be exhaustive in this part, but we aim to give a general view of how mainframe-like architecture can reduce the consumption of some specific context.

A centralized software architecture can improve the overall consumption of an application. In [21], the authors detail the recent software paradigm of micro-services, which seems to be the new standard. The idea to separate an application into multiple smaller elements allows more heterogeneous technologies, simplifies continuous integration, permits higher adaptability and flexibility. Artifacts are usually containerized and can run on distant servers. However, this model significantly increases communications computation because of the dependency of the artifacts on each other. In the case where these higher computations are not compensated by lower artifact consumption due to more appropriate technologies, the overall software consumption may increase.

With the example of a physical or virtualized server, a centralized architecture reduces the static consumption of the platform: one operating system consumes less than an identical one duplicated on other servers. The advantage of static consumption is compensated on some distributed architecture, but mainframe platforms would probably not benefit from it. By having databases and programs running at the same time, the mainframe reduces its computation, and the current usages do not require the use of technologies other than those present in a mainframe environment.

Another advantage is the limitation of the software bloat effect. The gradual addition of functionalities to an existing software often incurs a greater resource overhead in the same execution scenarios[27][28]. This is due to high-level programming where systems are based on layered modules. A typical scenario will use only a small percentage of the possibilities of a layer but will need to instantiate it anyway, leading with multiple layers to disproportionate consumption. We can argue that mainframe languages, especially for IMS and CICS, prevent this by avoiding frameworks and staying at relatively low-level programming. For example, an IMS application is typically composed of a set of compiled Cobol program which does not have to be instantiated from a higher perspective, there are simply called in a sequential chain. It is also commonly admitting that low-level program languages tend to consume fewer resources than higher ones[29].

4.2 Evolution of mainframe ecosystem

Despite its retro-compatibility, the mainframe ecosystem and the mainframe's place in the banking industry are evolving. Today, it is mostly used for historical applications when newer ones tend to be developed in a distributed environment. It is now interesting to study how the environment could evolve in the Information system (IS).

4.2.1 Virtualization perspectives: host or hosted

Rumors speaking about the decline of the mainframe can be found as early as the 90s[30] but it seems that the reality is more nuanced. The mainframe ecosystem has not disappeared, and we can argue that it won't be so in the next decade based on IBM's strategy. Its historical manufacturer keeps investing in it and mainframe sales represented $1.43 billion revenue in 2020.

Examples of companies with a mainframe migration record can be found, like Swisscom. The running mainframe code is moved to a virtualized mainframe, such as Software Defined Mainframe (SDM) by LzLabs. The software vendor presents it as a way to migrate applications without any recompilation. To the best of our knowledge, there is no independent performance review of this private solution, but the virtualization method degrades CPU performance due to the presence of static consumption of the host and virtualization software which is added to the consumption of the virtualized layer[31]. This performance degradation may be compensated by the gain in elasticity, which allows to adapt the host servers allocation and therefore the host servers consumption, but this is a strong assumption that needs a study in itself based on cost, quality of service, workload, and consumption. Finally, we can highlight an emerging trend where cloud providers such as Microsoft Azure and Google Cloud Platform are proposing modernization solutions to migrate mainframe workloads to their respective platforms.

Other companies continue to rely on physical mainframes. IBM takes the opposite direction from the hosted mainframe by offering new virtualization capabilities. For example, IBM LinuxONE servers are mainframes dedicated to hosting Linux virtual machines (VM). More traditional mainframes, such as IBM Z, can also host a Linux partition since 1999 in parallel with MVS LPARs. The latest version of z/OS can host containers and is able to schedule them with an orchestrator such as OpenShift. The performance degradation of these virtualization methods is probably less important since containers or VMs are virtualized, no matter who the host is. There is however a performance advantage to using containers instead of VMs due to their lightweight[32].

Both strategies do not answer the same objective, meaning that both will probably be adopted by different companies. The current proliferation of cloud services is a real shift in the industry. While companies may see an interest in migrating some of their applications to a public cloud and benefiting from its cost and performance efficiency, usually higher than that of a self-managed data center, they may also choose to develop a private cloud for their most sensitive data. This is an important architectural choice influenced by local jurisdiction, requirements, and available human resources. The mainframe shift may also not occur in some companies if the current architecture is conserved and if cloud solutions are not used or, for the private ones, based on distributed technologies.

4.2.2 Impact of skill shortage

The mainframe ecosystem has a skill shortage problem[33]. According to the specialized press, the majority of mainframe workloads are staying constant or increasing but, in some companies, up to 25% of the mainframe staff retired during the last five years. This trend is growing as only 7% of the mainframe workforce is under age 30. It can be explained by the fact that universities do not teach mainframe skills anymore, leading to a lack of knowledge but also interest from the young developer generation for the platform. In 2015, there were 1700 mainframe jobs opened up in the United States.

Some companies form their own staff to answer this problem. Partnerships with IBM are forged to create intern schools in companies under the "IBM Mainframe Academic Initiative", which facilitates the recruitment or the conversion of workers to the mainframe IT jobs needed. IBM also proposes an annual competition called "Master The Mainframe" to promote the environment along with students with challenges on a real platform.

The average age of mainframe staff is hard to estimate but it may be more than 50 years. The current turnover rate tends to show that there is a risk of global loss of competence when newer employees don't work long enough with experienced workers before their retirement. It is well documented that code written by inexperienced developers is less optimized than that of their colleagues[17] and we could see a decline in the mainframe performance in the coming years.

They are examples in the industry of companies addressing this issue by converting Cobol code to a more modern language, such as Java. Re-writing the applications by hand is, in most cases, too expensive but some automatic conversion can be done, but this solution currently has some problems. There is a performance issue between Java's floating-point-based number management and Cobol's fixed-point functions. Other performance issues may also occur due to the static conversion to a different programming paradigm. Finally, it is worth noticing that the migration of the mainframe application to a virtualized solution, or just in our case the code conversion, does not remove the needs for all mainframe jobs. Applications should still be maintained even if the platform changed, and a generated code is much less maintainable.

4.2.3 Banking ecosystem perspective

In the context of this study, a mainframe is a part of a more global banking Information system. It hosts some critical applications but also serves others acting like a back-end server. We identified three factors that currently impact or could impact the banking industry and therefore, the studied mainframe ecosystem. This section is incomplete as changes to the ICT and banking industry cannot be exhaustive, but it gives some identifiable influence.

Data engineering is a growing field in the industry. Having a well-designed big data architecture can provide a company with accurate metrics and trends about the market, which is an advantage relative to competitors. The challenges implied by this technology rely partially on accuracy and availability in a short time. By hosting part of the critical datasets of the companies, the mainframe has a role to play. [34] explains how a big data platform, such as Hadoop can be set up with a mainframe environment but architecture choice must be made due to the amount of transit data. A mainframe platform can perform a massive amount of I/O operations but exporting batch jobs is preferable to keeping a low treatment time for each request. Based on these needs, the mainframe architecture could evolve with, for example, running daemons as STC or as a containerized application to export the appropriate data.

Security needs are important in the banking industry. Mainframe compromise is rare due to their location in the architecture, an attacker would need to bypass many others security layers before accessing them, but this is not impossible. The role of security is also to protect company data from deliberated or non-deliberate destructive actions from employees. The mainframe environment is evolving to prevent an attack, which can have an impact on CPU consumption. For example, the most common mainframe security product will in the future propose to control the type of access made to the ACEE memory space by the different APF authorized programs. This security improvement may increase RACF CPU utilization due to the number of access made to the memory space. The generalization of default encryption on disks also increases CPU usage, which is hard to estimate due to the addition of dedicated cryptographic co-processors. It seems reasonable to assume that security needs are hard to conciliate with frugal objectives and that they tend to augment the resource platform resource consumption.

Finally, on a larger scale, we could also study the impact of cryptocurrencies on the global banking industry. By having a decentralized architecture, one of their main argument is the ability to use money without regulators and banks. However, some unresolved structural limits tend to prove that traditional money will, at least, cohabit with cryptocurrencies in the next decades. At the writing time, the most popular cryptocurrency is bitcoin, representing 55% of the total market capitalization. According to [35], the estimated annual miner's electricity consumption was 78.93TWh in 2019, a number which is now estimated at 125TWh by the Cambridge bitcoin electricity consumption index. Based on the 2019 consumption, this is equivalent to a two billion mainframe platform like the Groupe BPCE one. We can argue that banks are not limited to their mainframe platform, but they handle 90 percent of all credit card transactions. Moreover, due to the mining rate, the Bitcoin system is limited to seven transactions per second, which tends to make its common use currently impossible compared to the 2,000 operations handled on average by the Visa network. Cryptocurrencies may be used for other uses, such as speculation, but will not replace the traditional banking system in its current form and therefore will have no impact on mainframes.

4.3 Perspective of the mainframe on Green IT

The fast growth of information technologies has multiplied data centers and increased the carbon footprint of the field. Reducing hardware consumption is an area of research just like the emergence of energy-aware software[03]. Sustainable IT must be a global approach to limit the negative effect of IT expansion, with changes in product design, development, recycling, manufacturing, etc. In this part, we will study some of the relevant leads to integrate the mainframe into a sustainable approach.

4.3.1 Renewal of mainframes

Mainframes are regularly renewed. IBM proposes a new mainframe computer every two to three years: the z13 was launched in January 2015, the z14 in July 2017, and the z15 in September 2019. Each version improves the maximum number of processing units, the DRAM capacities, but they can also introduce new features such as hardware dedicated to cryptographic needs. Each generation may have a different declination and configuration.

Each generation seems to be rapidly adopted by banks. While we can't be sure about the equipment used by other companies, we can at least find an online trace of the latest three generations of the mainframe in Groupe BPCE and Credit Agricole. It is also quite complex to understand the motivation of renewal due to negotiation and contracts being usually secret between IBM and its clients. However, we can justify that it is not a necessity based on an increasing workload with our previous observation of the current z15 configuration: there are 172 processors available. We confirmed with the Groupe BPCE teams that the same order of magnitude of available processors could be found in the previous generation. While backup processors are necessary for high availability needs, the banking workload evolution could be managed without new mainframe materials.

Other parameters may be considered, such as the previously cited dedicated cryptographic hardware. However, it seems unlikely that the carbon footprint of the renewal will be taken into consideration because current IT architecture choices rarely study it. While the Manifesto for Energy-Aware Software argues that energy awareness should be engineered throughout the life cycle of an application and should be treated as an attribute of architectural quality[03], we can add that the carbon footprint of IT equipment must include the construction phase.

Some generation of mainframe servers, such as the z15, also implies the renewal of others equipment such as FICONs, tape library, and flash storage arrays to benefit from all the proposed improvements, leading to a shorter lifetime for this equipment. On the other hand, during the last two renewals, two mainframes out of the four changed were kept by Groupe BPCE to be used as backup servers for the recovery plan. This tends to extend the lifetime and it explains the three years of useful life we used in the carbon footprint estimation in part 3.2.2.

Based on the French electricity production mix, the most significant improvement of the mainframe carbon footprint must come from the manufacturing cycle. CO2 production linked to the mainframe consumption during its average 3-year lifetime is less important but could be improved by the PUE. Two other improvements are possible: increase the time between two renewals and choose a lighter configuration. The latest z15 configuration was proposed in five versions: Max34, Max71, Max108, Max145, and Max190. Each number corresponds to the number of processors. Four Max108 were chosen, while we saw that many processors are not used, despite the high availability constraint, leading to a higher static consumption but also a higher carbon production cost. If sustainable IT was taken into account along with other usual criteria, it could have led to other configurations such as an association of Max72 and Max108 processors.

4.3.2 Sustainable mainframe

We can now define what a sustainable mainframe platform may be. Based on our observation, mainframes efficiency attributes sometimes promoted by editors could not be verified. We justified that an x86 architecture with a similar workload and high availability criteria should have equivalent power consumption. Therefore, mainframes are not Green IT hardware elements by themselves but can integrate a sustainable strategy.

A sustainable mainframe strategy may be based on two axes. The first one would be the extension of the service life of mainframe servers, which reduces the most CO2 emitting phase in France, the manufacturing one. A mainframe renewal is an important architectural choice that influences platform equipment, and should therefore be based on clear objectives, including, but not only, sustainable criteria.

The second axis is code durability. Mainframe platforms have high retro-compatibility capabilities, allowing code to run for decades which reduces the need for technology-justified evolution. It reduces the carbon footprint of development and may have a significant impact on the platform efficiency. It also depends on the ecosystem's capability to form new developers as the platform needs to be maintained.

Finally, the evolution of mainframe and information technology is constant and may create new opportunities in the future. Technology intelligence, an activity that enables companies to identify opportunities and threats of the field may be required to benefit from the new gains of efficiency.

4.3.3 Limits

Because the mainframe platform is mostly closed source, it limits the capability of the ecosystem to be studied and/or improved. In our study, we made an approximation at each step which couldn't be verified easily and therefore, could not pretend that every mainframe platform would have an equivalent consumption to an x86 equivalent.

Based on the importance of the mainframes platform in the banking industry, studies should deepen into the efficiency of mainframe platforms, the efficiency of mainframe portability to x86 and the effect of code longevity.

Chapter 5

Conclusion

The fast growth of ICT increases the number of physical infrastructures and the overall consumption. Research on how to limit these effects concentrates on x86 architecture but tends to forget about specificities. With regard to its importance in the banking industry, we studied mainframe platforms based on the Groupe BPCE one.

In the first part, we saw how the usual consumption evaluation method could be declined. As a mainframe platform is a hypervisor, with its LPAR which can be compared to a VM, the most practical way is to use a software approach. In this model, LPAR metrics would be extracted and used in a model to estimate consumption. Compared to the hardware or hybrid approach, it is easier to put in place, and it allows for finer granularity at the cost of lower precision.

In the second part, we analyzed the CPU consumption of the mainframe servers as they are the most common metric used in the software model. We realized that the mainframe server consumption to be mostly static as the processors have a fixed frequency. We also justified with a global consumption study of the platform that CPU is not the most consuming element as mainframes servers have lower consumption than storage elements. Therefore, we justified that the traditional evaluation of software consumption could not be applied. The Groupe BPCE mainframe platform is estimated to consume about 150 kW or 243 kW if we take into account the datacenters PUE. Combined with its manufacturing emission and its lifetime, we determined an annual carbon footprint of 311 tons eq. CO2 in France. It highly depends on the electricity production method of the host country, as the carbon footprint is about 1,345 tons eq. CO2 in the United States and about 1,769 tons eq.

CO2 in China.

We used mainframe MIPS to determine the size of an equivalent x86 platform. The Groupe BPCE mainframe servers theoretical equivalent uses 308 servers. It would have a similar electrical consumption compared to mainframe servers and a lower carbon footprint based on the average server lifetime which is higher than the mainframe server lifetime observed at Groupe BPCE. The manufacturing footprint was estimated by a simplistic computation based on weight and should be replaced by actual values.

In the third part, we detail how a sustainable IT strategy can be declined to a mainframe platform. An extension of equipment lifetime could be accompanied by a software strategy. The optimization of the interactive and batch transactions has a limited effect, due to the high static consumption and low-level programming, but the retro compatibility capacity of the platform could allow a lower carbon footprint of development, which remains to be quantified. The evolution of the mainframe ecosystem evolution can take place as a VM/container or it can be hosted, we think both choices should be made with sustainable IT criteria in mind. The ecosystem can also be impacted by external factors such as skill shortage, security, and big data needs. All these factors could increase the consumption of mainframes in a way that should be studied and compared to x86 architectures.

We conclude by the need to further study mainframe platforms as they are responsible for most of the current financial transactions. This study is based on the Groupe BPCE mainframe platform and should be completed with metrics from other systems. The consumption of non-mainframe server elements, such as tape libraries and flash storage arrays, should be detailed as they were mostly presumed from peak consumption. Finally, an equivalent x86 platform should be tested in real conditions to confirm the assumptions made in this study.

List of Figures

Bibliography

[01] Gobinda Chowdhury. How digital information services can reduce greenhouse gas emissions. *Online Information Review*, 36, 08 2012.

[02] Jens Malmodin and Dag Lundén. The energy and carbon footprint of the global ict and e/m sectors 2010–2015. *Sustainability*, 08 2018.

[03] Alcides Fonseca, Rick Kazman, and Patricia Lago. A manifesto for energy-aware software. *IEEE Software*, 36(6):79–82, 2019.

[04] Dost KHAN and Nawaz Mohamudally. From mainframe to cloud computing: A study of programming paradigms with the evolution of client-server architecture. *Journal on Computing*, vol. 3,:pp 21–27, 12 2011.

[05] Mohammad Amin, Md Alauddin, and Dr Azad. Business transaction processing system. *International Journal of Computer Information Systems*, 4:11–15, 05 2012.

[06] IBM. *MVS System Management Facilities(SMF)*. International Business Machines Corporation, New York, NY, USA, 7 2019. IBM z/OS 2.3 Manual.

[07] Dimitris Tsirogiannis, Stavros Harizopoulos, and Mehul A. Shah. Analyzing the energy efficiency of a database server. In *2010 ACM SIGMOD International Conference on Management of Data*, SIGMOD '10, page 231–242, New York, NY, USA, 2010. Association for Computing Machinery.

[08] Zichen Xu, Yi-Cheng Tu, and Xiaorui Wang. Pet: Reducing database energy cost via query optimization. *Proc. VLDB Endow.*, 5(12):1954–1957, August 2012.

[09] Rafael Aroca and Luiz Gonçalves. Towards green data centers: A comparison of x86 and arm architectures power efficiency. *Journal of Parallel and Distributed Computing*, 72:1770–1780, 12 2012.

[10] Huazhe Zhang and Henry Hoffmann. Maximizing performance under a power cap: A comparison of hardware, software, and hybrid techniques. In *Twenty-First International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '16, page 545–559, New York, NY, USA, 2016. Association for Computing Machinery.

[11] Ulysse Rugwiro and Gu Chunhua. Customization of virtual machine allocation policy using k-means clustering algorithm to minimize power consumption in data centers. In *Second International Conference on Internet of Things, Data and Cloud Computing*, ICC '17, New York, NY, USA, 2017. Association for Computing Machinery.

[12] IBM. *IBM z15 (8561) Technical Guide*. International Business Machines Corporation, New York, NY, USA, 8 2020. IBM Redbooks.

[13] Hayri Acar. *Software development methodology in a Green IT environment*. Theses, Université de Lyon, November 2017.

[14] Kwame Chan Jong Chu, Tanjina Islam, Miguel Exposito, Sanjay Sheombar, Christian Valladares, Olivier Philippot, Eoin Grua, and Ivano Malavolta. Investigating the correlation between performance scores and energy consumption of mobile web apps. pages 190–199, 04 2020.

[15] Guillaume Fieni, Romain Rouvoy, and Lionel Seinturier. Smartwatts: Self-calibrating software-defined power meter for containers. In *CCGRID 2020 - 20th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing*, Melbourne, Australia, May 2020.

[16] Wentai Wu, Weiwei Lin, and Zhiping Peng. An intelligent power consumption model for virtual machines under cpu-intensive workload in cloud environment. *Soft Computing*, 21, 10 2017.

[17] Eva Kern, Markus Dick, Stefan Naumann, and Tim Hiller. Impacts of software and its engineering on the carbon footprint of ict. *Environmental Impact Assessment Review*, 52, 08 2014.

[18] Juha Taina. How green is your software? In Pasi Tyrväinen, Slinger Jansen, and Michael Cusumano, editors, *Software Business*, volume 51, pages 151–162, United States, 2010. Springer. Volume: Proceeding volume: 51; ICSOB 2010 ; Conference date: 21-06-2010 Through 23-06-2010.

[19] Baihe Gu, Xianchun Tan, Yuan Zeng, and Zekun Mu. Co2 emission reduction potential in china's electricity sector: Scenario analysis based on lmdi decomposition. *Energy Procedia*, 75:2436–2447, 08 2015.

[20] Kerstin Hötte, Anton Pichler, and François Lafond. The rise of science in low-carbon energy technologies. *Renewable and Sustainable Energy Reviews*, 139:110654, 04 2021.

[21] Yu Gan, Yanqi Zhang, Dailun Cheng, Ankitha Shetty, Priyal Rathi, Nayan Katarki, Ariana Bruno, Justin Hu, Brian Ritchken, Brendon Jackson, Kelvin Hu, Meghna Pancholi, Yuan He, Brett Clancy, Chris Colen, Fukang Wen, Catherine Leung, Siyuan Wang, Leon Zaruvinsky, Mateo Espinosa, Rick Lin, Zhongling Liu, Jake Padilla, and Christina Delimitrou. An open-source benchmark suite for microservices and their hardware-software implications for cloud and edge systems. In *Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS '19, page 3–18, New York, NY, USA, 2019. Association for Computing Machinery.

[22] HP. *Mainframe Alternative Sizing*. Hewlett-Packards.

[23] TmaxSoft. *TmaxSoft OpenFrame and Microsoft Azure*. TmaxSoft.

Abstract: Mainframe Migration Sizing Study

[24] Md Sabbir Hasan, Frederico Alvares De Oliveira, Thomas Ledoux, and Jean-Louis Pazat. Enabling Green Energy awareness in Interactive Cloud Application. In *IEEE International Conference on Cloud Computing Technology and Science 2016*, Luxembourg, Luxembourg, December 2016.

[25] Georgiana Copil, Daniel Moldovan, Ioan Salomie, Tudor Cioara, Ionut Anghel, and Diana Borza. Cloud sla negotiation for energy saving: A particle swarm optimization approach. pages 289–296, 08 2012.

[26] Olli Mämmelä, Mikko Majanen, Robert Basmadjian, Hermann Meer, André Giesler, and Willi Homberg. Energy-aware job scheduler for high-performance computing. *Computer Science - Research and Development*, 27, 11 2012.

[27] Suparna Bhattacharya, Kanchi Gopinath, Karthick Rajamani, and Manish Gupta. Software bloat and wasted joules: Is modularity a hurdle to green software? *IEEE Computer*, 44:97–101, 09 2011.

[28] Harry Xu, Nick Mitchell, Matthew Arnold, Atanas Rountev, and Gary Sevitsky. Software bloat analysis: Finding, removing, and preventing performance problems in modern large-scale object-oriented applications. pages 421–426, 01 2010.

[29] Rui Pereira, Marco Couto, Francisco Ribeiro, Rui Rua, Jácome Cunha, João Paulo Fernandes, and João Saraiva. Energy efficiency across programming languages: How do energy, time, and memory relate? In *10th ACM SIGPLAN International Conference on Software Language Engineering*, SLE 2017, page 256–267, New York, NY, USA, 2017. Association for Computing Machinery.

[30] Stewart Alsop. Ibm still has the brains to be a player in client/server platforms. *InfoWorld*, 03 1993.

[31] Javier Cacheiro, Carlos Sánchez, Esteban Freire, Sergio Díaz, and Alvaro Simón. Providing grid services based on virtualization and cloud technologies. volume 6043, pages 444–453, 08 2009.

[32] Roberto Morabito, Jimmy Kjällman, and Miika Komu. Hypervisors vs. lightweight virtualization: A performance comparison. 03 2015.

[33] Thomas Ngo-Ye and Jae Choi. Teaching students mainframe skills for the niche market: An exploratory proposal. 03 2018.

[34] Cameron Seay, Rajeev Agrawal, Anirudh Kadadi, and Yannick Barel. Using hadoop on the mainframe: A big solution for the challenges of big data. pages 765–769, 05 2015.

[35] Kateryna Kononova and Anton Dek. Bitcoin carbon footprint: Mining pools based estimate methodology. urn:nbn:de:0074-2761-0:265–273, 12 2020.