



# jamk

## zManav

### Db2 for z/OS Wellness Monitor

Manigandan Ravichandran

Master's thesis

Nov 2024

Master's Degree Programme in Artificial Intelligence and Data Analytics

**jamk** | Jyväskylän ammattikorkeakoulu  
University of Applied Sciences

**Ravichandran, Manigandan**

**zMANAV-Db2 for z/OS Wellness Monitor**

Jyväskylä: Jamk University of Applied Sciences, November 2024, 54 pages.

Degree Programme in Information Technology, Artificial Intelligence and Data Analytics. Master's thesis.

Permission for open access publication: Yes

Language of publication: English

**Abstract**

The zManav, Db2 for z/OS Wellness Monitor, is an AI-driven solution that automates the monitoring and analysis of Mainframe Db2 for z/OS operational logs. The development research addresses the challenge of analyzing complex mainframe syslogs, which are prone to anomalies during high workload periods, disaster recovery, during major mainframe infrastructure h/w and s/w upgrade and maintenance activities.

By leveraging periodic automated reporting from Db2 for z/OS MSTR JESMSLG integrated with graphical visualizations, the zManav streamlined log analysis and enhanced administrative efficiency.

Using Design Science Research method, the author developed an automated syslog parsing and anomaly detection tool-zManav which tailored to Db2 MSTR JESMSGLG logs. The Design Science research method is categorised into five phases. They are as follows:

- (i) Problem and Motivation: Transform humanistic monitoring to AI based monitoring and alerting.
- (ii) Objective Definition: Develop a technique to transform logs to structural format for data analytics.
- (iii) Design and Development: Using pandas python libraries in Databricks, produce metrics and alerts.
- (iv) Demonstration: Demonstrate the metrics in time-series analysis framework using zManav tool.
- (v) Communication: Document the research development as Master's thesis and training to the team.

The AI-driven analysis and monitoring process involved extracting relevant data from Mainframe Db2 system logs and moved to Databricks as text files for Data analytics using Pandas Data Frame and the processed data is then stored as TSV (Tab Separated Value) format. The resulting metrics using Data analytics, facilitated behaviour analysis by identifying trends associated with selected message IDs and highlighting new message IDs that might require attention.

The zManav not only streamlined the monitoring and analysis of mainframe Db2 logs but also established a foundation for smarter, data-driven mainframe management. By automating routine checks, enhancing predictive capabilities, and fostering proactive responses, the tool demonstrated its potential to transform mainframe administration practices and improve overall operational efficiency.

**Keywords/tags (subjects)**

Mainframe, Db2 SYSLOGS, TSV file format, Message ID, Databricks, Pandas Data Frame

**Miscellaneous (Confidential information)**

-

## Acknowledgements

I would like to express my heartfelt gratitude to my spouse Mrs. Nireesha Manigandan (MSc. Biochemistry) and my children's Vijayendra Sai Manigandan and Raghavendra Sai Manigandan for their unwavering motivation and support throughout my academic journey.

I would also like express my deepest gratitude to my thesis advisor, Dr. Sipola Tuomo, and our Principal Lecturer Mr. Mika Rantonen and our Student Advisor/Senior Lecturer Ms. Tuula Kotikoski for their unwavering guidance, support, and encouragement throughout the course of this research.

My heartfelt thanks also go to my family and friends, whose patience and understanding made this journey possible. I am also grateful to all those who assisted me with the academic resources and technical support, including Jukka Köykkä (OP DB Product Owner, OP BANK), Sallamaari Tervakangas (Competence lead, Specialists Competence Area, OP BANK), Robinder Singh (Senior Db2 and IMS System Engineer), Ramkumar Sundararajan (Growth Marketing Data Analyst, Swappie), Dinesh Kumar Damodharan (Senior Consultant, Web/API Automation, Capgemini Sogeti), Anbalagan Chitraprasanna (Senior Specialist, Chapter Lead, OP BANK). Their contributions have been invaluable to my academic success, and I am forever grateful for their help and encouragement.

Sincere thanks to OP software academy training initiatives on the Databricks and Analytics session. It opened the doors for me to implement my ideas in a standard platform. The sessions on Databricks and analytics equipped me with the skills to conduct my research and successfully complete my thesis.

## Contents

<b>1</b>	<b>Introduction .....</b>	<b>6</b>
1.1	Mainframe technology in Finnish institutions .....	7
1.2	Research Motivation .....	7
1.2.1	Mainframe Talent: Addressing the Looming Talent Shortage .....	9
1.2.2	Why zManav standsout : AI-Driven Syslog Analysis for Mainframe Monitoring ..	10
1.2.3	zManav bridging the gap .....	11
1.3	Research questions .....	11
<b>2</b>	<b>Research Methodology .....</b>	<b>12</b>
2.1	Problem & Motivation.....	12
2.2	Objective Definition .....	13
2.3	Design & Development .....	13
2.4	Demonstration & evaluation.....	13
2.5	Communication .....	14
<b>3</b>	<b>Research Ethicality.....</b>	<b>14</b>
3.1	AI on demand .....	14
3.2	Research copyright.....	15
3.3	Scientific Research References .....	15
<b>4</b>	<b>Research Reliability .....</b>	<b>15</b>
4.1	The reliability of the input file.....	15
4.2	The reliability of the python techniques.....	16
4.3	The reliability of the Output file.....	16
4.4	The reliability of the data platform.....	16
4.5	The reliability of the results .....	17
<b>5</b>	<b>About zSystem - Mainframe.....</b>	<b>17</b>
5.1	About Mainframe z facts.....	18
5.2	Db2 for z/OS architecture .....	19
5.3	About Mainframe Db2 System Programmers and responsibilities .....	19
5.4	BAU vs Major Upgrade challenges .....	20
5.5	Db2 sysprog BAU challenges .....	20
5.6	Db2 sysprog Upgrade challenges .....	21
<b>6</b>	<b>Thesis Design and development.....</b>	<b>22</b>
6.1	Thesis Research strategy.....	23

<b>7</b>	<b>Results : Thesis research outcome.....</b>	<b>29</b>
7.1	DSNJ002I Full Active Log .....	29
7.2	DSNJ139I OFFLOAD TASK ENDED.....	30
7.3	DSNQ022I MQ listener Failed to connect. ....	31
<b>8</b>	<b>Developments .....</b>	<b>31</b>
8.1	Tool as user interactive .....	32
8.2	Focused Trend Analysis approach using user interactive options.....	33
8.2.1	DSNJ031I WARNING - UNCOMMITTED UR HAS WRITTEN number LOG RECORDS.	36
8.2.2	DSNB260I WARNING - A READER HAS BEEN RUNNING FOR duration MINUTES..	37
8.2.3	Focused Trend Analysis approach on Db2 for z/OSGroup Buffer Pool Shortages	37
8.2.4	Trend of DSNB319A (Shortage), DSNB325A (Critical), DSNB327I (Adequate) .....	40
<b>9</b>	<b>Challaneges and discussions.....</b>	<b>44</b>
9.1	Steps incorporated to tackle challenges in source data analysis.....	44
9.2	Limitations with in Databricks.....	45
9.3	Further research expansions.....	46
9.4	Mainframe Root Cause Analysis.....	47
<b>10</b>	<b>Conclusions .....</b>	<b>48</b>
	<b>References .....</b>	<b>50</b>
	<b>Appendices .....</b>	<b>53</b>
	Appendix 1. Data Management Plan .....	53
 <b>Figures</b>		
	Figure 1 : Design science research method .....	12
	Figure 2 Research Design .....	23
	Figure 3 REXX JESMSGLG extraction .....	26
	Figure 4 FTP selection in Mainframe .....	26
	Figure 5 Merge Continuation Lines.....	27
	Figure 6 Parse to structured columns.....	28
	Figure 7 Save Dtaframe as TSV .....	29
	Figure 8 DSNJ002I Full Active log .....	30
	Figure 9 DSNJI139I LOG OFFLOAD TASK ENDED.....	31
	Figure 10 DSNQ002I MQ LISTNER FAILED.....	31
	Figure 11: User Interactive technique .....	33
	Figure 12 User Interactive Db2 Selection .....	35

Figure 13 User Interactive Message ID Selection .....	35
Figure 14 DSNJ031I WARNING -UNCOMMITTED UR HAS WRITTEN number LOG RECORDS ....	36
Figure 15 DSNB260I WARNING - A READER HAS BEEN RUNNING FOR duration MINUTES.....	37
Figure 16 Group Buffer Alert Pool Sum Total .....	39
Figure 17 Group BufferPool trend technique .....	39
Figure 18 GBP5 TREND.....	40
Figure 19 GBP5-SEC TREND.....	41
Figure 20 GBP4 TREND.....	41
Figure 21 GBP4-SEC TREND.....	42
Figure 22 GBP0 TREND.....	42
Figure 23 GBP0-SEC TREND.....	43
Figure 24 GBP SHORTAGE overall report.....	44
Figure 25 Challenges with archive log readers .....	45
Figure 26 DSNT360I with 35 asterisks DISPLAY REPORT.....	45
Figure 27 TSV file size limit exceeded.....	46
Figure 28 Bad request unable to export TSV .....	46

## Tables

Table 1: Platform Readiness .....	24
-----------------------------------	----

## 1 Introduction

Mainframes, recognized as the most robust computing systems, play a crucial role in global business and technology. They are utilized by 71% of Fortune 500 companies and handle 90% of all credit card transactions. Additionally, they manage 69% of the world's production IT workload (Precisely, 2022). Beyond their commercial use, mainframes have significantly contributed to space research, from powering the Apollo 11 mission in 1969 (Apollo11Space, n.d.) to supporting India's Chandrayaan-3 lunar rover in 2023 (ZMainframes, n.d.).

In this thesis, the primary objective is to address the challenge of analyzing mainframe syslogs, particularly focusing on trend of the message IDs to detect anomalies in Db2 MSTR JESMSGLOG logs during high workload scenarios and alert on new message ID that has never occurred before. To achieve this, a structured Design Science Research (DSR) approach is employed. The work devel-

ops an automated syslog parsing and anomaly detection tool, augmented by a dynamic "keep dictionary" technique that flags new messages and enables expert validation. The outputs of this research include a functional artifact for syslog analysis, design principles for handling high-volume logging systems, and explanatory insights into the patterns of workload-induced anomalies.

The contribution of this thesis lies in providing a scalable and adaptive method for syslog analysis, enhancing the ability to identify and trend the system behaviour up on workload and determine anomalies in real-time. This work provides a practical tool that enhances operational efficiency by eliminating the need for traditional manual syslog processing.

## **1.1 Mainframe technology in Finnish institutions**

As an example, I would like to highlight Kela, Finland's social insurance institution, has made a significant move towards modernizing its IT infrastructure by acquiring central computing technologies from IBM, valued at approximately €60 million. This decision reflects Kela's commitment to enhancing the efficiency and scalability of its services, especially in handling vast amounts of data related to Finland's social welfare programs. The collaboration with IBM underscores Kela's dedication to leveraging cutting-edge technology to improve service delivery, streamline operations, and ensure that the institution remains agile and responsive to the needs of Finnish citizens. The investment is part of a broader trend in both the public and private sectors to integrate more robust, flexible, and secure technological solutions into critical government services (Kela 2024, November 21).

## **1.2 Research Motivation**

With over 18 years of experience as a mainframe infrastructure administrator, I have gained an in-depth understanding of the complexities and critical nature of mainframe operations and system administration. Over the years, I have observed numerous challenges in monitoring Db2 for z/OS systems and have often envisioned a solution that could streamline this process while enhancing operational reliability.

This experience has inspired the main goal of my thesis: the development of a tool named "zManav." This tool is intended to simplify the challenges faced by mainframe administrators in an

era where data is as valuable as oil. IBM's Db2 for z/OS, alongside IBM IMS, offers a highly reliable, secure, and scalable database environment on the mainframe platform, and is the most used database in this context (IBM, n.d.-a).

The lowercase 'z' in "zManav" is deliberately chosen to align with IBM's naming conventions, where 'z' signifies 'zero downtime,' reflecting the tool's aim to enhance system reliability and availability (IBM Developer, n.d.). "Manav," derived from the Sanskrit word for "student" or "human," represents the tool's design philosophy (Sanatana Dhara, n.d.). It embodies the concept of a learning, intelligent entity within the artificial intelligence paradigm, akin to a student or human learning and adapting in the realm of mainframe technology.

zManav will be a revolutionizing mainframe monitoring tool with automated Db2 syslog checks. It focuses on the software aspect of the Db2 subsystem, excluding customer or application data, to address issues such as deadlocks, timeouts, security violations and resource unavailability, new critical alert and so on. Through periodic automated reporting facilitated by a graphic tool, zManav aspires to streamline and improve the efficiency of mainframe administration. The thesis also incorporates dictionary technique from previously generated message and message ID to recognize patterns and identify new system messages. Additionally, it provides alerts when new message arrive that may be system-critical and require potential action from system programmer.

The development of zManav stands as a significant milestone in the field of Db2 systems management, especially in environments where overseeing multiple subsystems is a common task. For shops managing over 20 Db2 subsystems, the daily manual checks required for each system can be exceptionally time-consuming and labor-intensive. The introduction of zManav is set to bring substantial relief to this process. By automating these routine checks, zManav not only streamlines the workflow but also allows Db2 systems programmers to redirect their focus to more critical tasks, such as Business As Usual (BAU) activities and intricate upgrade projects. This shift from manual to automated monitoring ensures more efficient identification and handling of potential issues in each subsystem, enhancing proactive management.

Moreover, the author, recognizing the broader potential of this tool, has plans to expand its capabilities beyond Db2 subsystems and implement the same for below Mainframe software subsystems.

- MQ (Message Queue) for z/OS.
- IMS (Information Management System database) for z/OS.
- CICS (Customer Information Control System) for z/OS.
- z/OS (z Mainframe Operating System syslogs)

The vision is to adapt and evolve zManav for monitoring all other subsystems within the Mainframe platform. This expansion would further consolidate the tool's utility, making it a comprehensive solution for Mainframe monitoring. Such development could significantly enhance operational efficiency, ensuring that Mainframe administrators are equipped with a versatile tool that caters to a wide array of monitoring and management needs across the Mainframe landscape. This forward-thinking approach, enhanced by the integration of user interactive data analytics monitoring techniques, underscores the author's commitment to continual improvement and adaptation in the rapidly evolving field of Mainframe technology (Business Research Insights, 2024).

### **1.2.1 Mainframe Talent: Addressing the Looming Talent Shortage**

While Mainframes still remain to be a critical backbone for many industries, including banking, insurance, healthcare, and government, processing over 70% of the world's online transactions daily (Deloitte, n.d.-a). However, the growing reliance on these systems is increasingly at risk due to a looming talent shortage. As seasoned professionals approach retirement, the availability of qualified mainframe specialists diminishes, creating a knowledge gap that could threaten operational efficiency and security. Addressing this challenge requires proactive measures such as upskilling current IT staff, integrating mainframe education into university curriculums, and leveraging automated tools to reduce dependency on manual processes. Organizations must also modernize their recruitment strategies to attract younger talent and demonstrate that mainframe expertise remains a valuable and evolving career path in the digital age.

Well renowned Deloitte (Deloitte, n.d.-b), largest global professional services firms, offering expertise in areas such as consulting, audit, tax, financial advisory, and risk management provides insights into the challenges faced by organizations relying on mainframes due to a declining talent pool (Deloitte, n.d.-a). It discusses:

- The critical role of mainframes in modern IT infrastructure.
- Strategies to address the talent gap, such as upskilling existing employees and recruiting new talent.
- The generational shift in mainframe professionals as retirements increase.
- Collaborative efforts between corporations, educational institutions, and training programs are essential to bridge this gap and ensure the sustainability of critical mainframe operations.

Refer, e.g., Yle Uutiset (2024,3) where Nordea, one of the largest financial services groups in the Nordic region, headquartered in Helsinki, Finland, with a significant presence across Sweden, Denmark, and Norway rely on legacy mainframe technology for critical banking operations and the recent issue demonstrates the criticality and operational challenges and measures.

### **1.2.2 Why zManav standsout : AI-Driven Syslog Analysis for Mainframe Monitoring**

While there are numerous well-established, cost-oriented, licensed mainframe log monitoring tools, such as Dynatrace, IBM Z Operational Log and Data Analytics, Splunk for Mainframe and BMC MainView, etc., zManav stands out for its simplicity and independence. Unlike these third-party vendor solutions, which require ongoing licenses and integrations, zManav leverages existing resources within the author's organization, it is license-free, and operates independently without reliance on external vendors.

The use of advanced data analytics in syslog analysis is another key differentiator. Traditional monitoring tools focus on performance metrics and alerts, but zManav tool goes a step further by applying advanced analytics to syslogs, enabling more precise anomaly detection and trend forecasting. This approach not only improves the efficiency of monitoring but also empowers organizations with the ability to proactively address potential system issues before they escalate.

Furthermore, by utilizing AI, the tool offers an easy-to-deploy solution that can continuously evolve with the growing needs of mainframe systems, without requiring complex vendor configurations or costly software updates. This makes it a cost-effective and sustainable choice for businesses looking to maintain robust, reliable mainframe operations in-house.

Thus, zManav provides a unique, efficient, and vendor-independent alternative for mainframe system log behaviour analysis, making it especially suitable for organizations looking for high-performance monitoring without the burden of external licenses or vendor dependency.

### **1.2.3 zManav bridging the gap**

In the face of resource crises, such as the shortage of skilled mainframe staff, and the increasing need for efficient monitoring during business-as-usual (BAU) operations, disaster recovery (DR), and regulatory compliance (specifically for DORA), a tailored AI-driven monitoring tool can significantly bridge the gap. By automating critical processes like log analysis, alerting on new message IDs, and ensuring the continuous health of mainframe systems, the tool can offer real-time insights and proactive incident management, thereby alleviating the challenges posed by human resource shortages. Moreover, the tool's ability to run both during regular operations and in disaster recovery scenarios ensures uninterrupted monitoring and reporting, thus maintaining compliance with the Digital Operational Resilience Act (DORA). This innovation not only addresses the evolving demands of mainframe management but also guarantees operational resilience, crucial in an era of growing cyber threats and regulatory oversight (EBA, n.d.)

## **1.3 Research questions**

Specifically, this study seeks to answer the following research questions:

(RQ1) How can the analysis of Db2 for z/OS system logs be automated to enhance the detection of critical message IDs and prevent operational disruptions?

(RQ2) What role can lookup techniques, such as dictionary-based methods, play in identifying new and potentially critical message IDs?

(RQ3) How can trend analysis using time series data improve the proactive and reactive management and situational awareness of Db2 subsystem behaviours?

By addressing these questions, this research aims to bridge the gap in current mainframe monitoring practices, ensuring operational continuity while safeguarding the organization's reputation.

## 2 Research Methodology

The centerpiece of my master's degree thesis, zManav, aims to revolutionize how Mainframe Db2 system behaviour is monitored and analyzed from the operational logs using design science research method (Peffer et al., 2006). Refer (Figure 1). This Process comprises of 5 distinct steps, namely Problem & Motivation, Objective definition, design and development, demonstration and communication. Each of these steps is described in detail in the subsequent paragraphs.

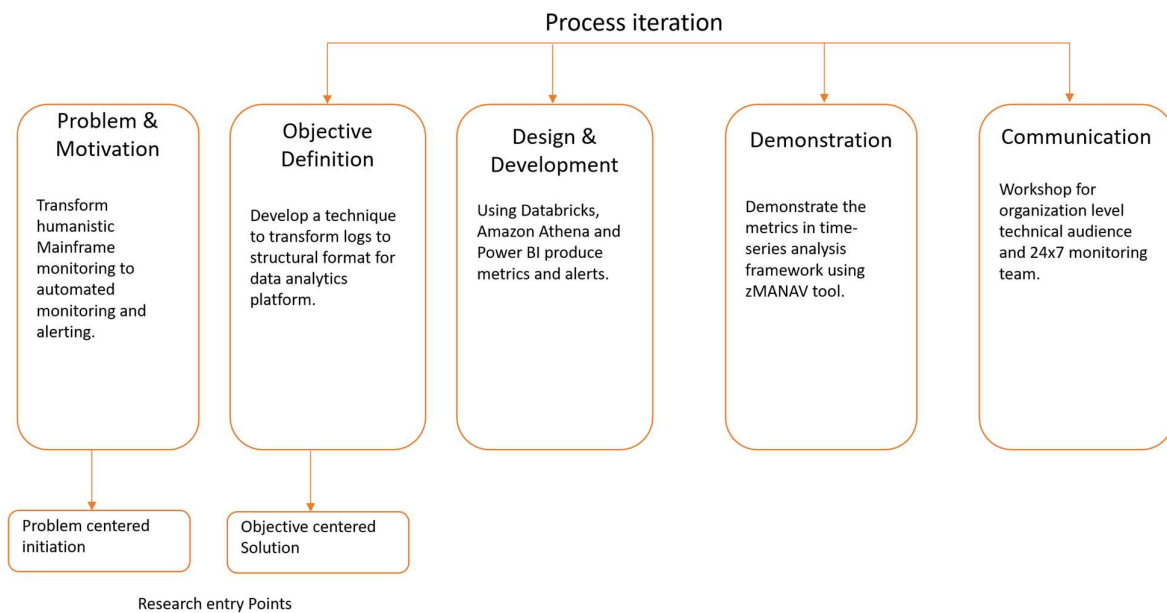


Figure 1 : Design science research method

### 2.1 Problem & Motivation

The development of an autonomous data analytics based monitoring system for monolithic mainframe computers has been driven by several critical challenges. These include the shortage of skilled labour in high demand, declining interest among new talent in learning mainframe technologies, the need to manage continuous software and hardware upgrades, and the risks posed by human errors, such as missed alerts on the mainframe console. Additional pressures include regulatory compliance requirements like enforcement of the Digital Operational Resilience Act (DORA) introduced by the European Banking Authority (EBA, n.d.), escalating geopolitical tensions, increasing cyber threats, and the persistent demand for mainframe reliability and performance.

## **2.2 Objective Definition**

The primary objective is to eliminate the need for manual scanning of Db2 subsystem logs (JESMSGLOG- Details are furnished in the below upcoming section) to assess operational behaviour. Instead, the aim is to transform these logs into a structured format suitable for data analytics. This process will enable the generation of trend reports using metrics collected from the system logs, thereby improving efficiency and operational insights.

## **2.3 Design & Development**

Comprehensive overview of the system's operational behaviour is achieved by leveraging advanced analytics to enhance decision-making and system reliability.

The proposed design involves developing a robust technique capable of operating on a highly reliable platform such as Databricks. This approach will process data to populate trend and alert reports using Power BI, with Amazon Athena serving as the backend database.

To ensure resilience, the technique must also function as a standalone solution that can operate independently on local laptop machines. This contingency feature would allow uninterrupted reporting and monitoring in the event of a major disaster or service interruption affecting cloud-based systems.

## **2.4 Demonstration & evaluation**

The system will showcase the metric trends through time-series analysis, providing a "helicopter view" of the overall subsystem's health and behaviour. This includes automated alerting for any newly encountered message IDs that have not appeared in historical records. While trend evaluation traditionally relies on rich technical expertise, this approach allows users to gain insights into system behaviour. The visualization of these trends ensures accessible and actionable monitoring for both experienced and less-experienced users.

## 2.5 Communication

The thesis itself serves as a medium to document and communicate the innovation, including its design process and development journey. In addition, conducting workshops on the overall reporting model and the capabilities of the proposed technique will foster a deeper understanding and appreciation of the research approach. This initiative will significantly enhance the performance and efficiency of the 24x7 Mainframe Operations and Monitoring teams. Additionally, an automated alerting mechanism, which sends email notifications whenever a new message ID is detected, is currently under development to further streamline monitoring processes.

## 3 Research Ethicality

Appropriate approvals are secured from OP bank database product owner and competence lead to undertake the thesis using the Mainframe Db2 software logs from Non-Production environment. In this particular case, I will be focusing on the Mainframe Db2 logs and the techniques will be further developed for other subsystems including z/OS syslog's. Due to organization's Data Protection regulations, the input logs or processed output logs shall not be saved in any personal media, public domain or public cloud other than OP specific intranet storage medium. The thesis is performed using TEST/DEV (Non-prod) software logs and when matured, the author shall decide and deploy it considering the techniques for production implementation that is limited for OP bank use only.

No personal data is available in Mainframe software logs and not applicable in this research development of zManav.

### 3.1 AI on demand

Human intelligence and thought processes are incomparable to artificial intelligence. However, I find AI-based models increasingly in demand for assisting humanity in various ways. I would like to acknowledge the use of Artificial Intelligence (AI) for choosing better words in this thesis planning document. I have used AI for grammatical corrections and suggest alternative words that better convey my intended meaning. The ideas presented in this thesis planning document are my own, and the use of AI was limited to grammatical corrections and suggesting better word choices only.

In my research, AI tools like ChatGPT and Copilot have been invaluable. They helped resolve logical code-level problems in error solving process as well.

### **3.2 Research copyright**

Further patents / IPR related to the zManav research technique or zManav research model to be discussed within OP organization IPR/innovation lab team.

As per author's employment agreement with Company OP Osuuskunta, any research work conducted during my tenure, whether directly related to my duties or otherwise, is considered the intellectual property of the company. This includes all findings, inventions, and developments produced in connection with my work for the organization.

OP bank owns copyrights (Intellectual Property Rights) created or developed by the Employee/Executive, either alone or jointly with others, during the course of his/her employment in OP Financial Group and while performing duties based on the employment or otherwise in connection with his/her employment.

### **3.3 Scientific Research References**

Due to the industry-driven nature of mainframe computing, much of the research on this topic is found in case studies, legacy documentation, and specialized blogs, rather than in academic or peer-reviewed scientific journals. As a result, there is a limited availability of scientific research articles focusing on mainframe systems. The author has endeavoured to review and integrate the available research into this thesis, while recognizing that much of the literature in this field is primarily found in blogs and industry-focused publications rather than peer-reviewed scientific journals.

## **4 Research Reliability**

From the research perspective, the zManav research reliability is categorized into five aspects.

They are as follows in subsection:

### **4.1 The reliability of the input file**

The Mainframe Db2 JESMSGLG syslog that are in scope is considered as input. They are generated by default by Db2 subsystem as they are started in Mainframe and available throughout its lifecycle (Until They are shutdown). The format of Db2 JESMSGLG in mainframe environments, which records subsystem-related activities, remains integral to the operational behaviour of z/OS and Db2 systems. While the core structure of JESMSGLG is unlikely to change significantly, updates in

Db2 and z/OS, such as new function levels and enhancements, could influence how these logs are processed and analyzed. For example, IBM's continuous delivery model for Db2 introduces new capabilities, but maintains backward compatibility, ensuring that the underlying log structure remains consistent while offering improvements in performance and functionality (IBM, n.d.-b).

## 4.2 The reliability of the python techniques

For the purpose of research technique, we have used Pandas libraries, The re (Regular Expression) and the matplotlib. These libraries receive updates, but its core functionality remains stable. The improvements tend to focus on additional features, new plot types, and performance optimizations. and let's look at the reliability of each one of them individually.

(i) Pandas which is one of the most widely used libraries for data manipulation. It allows for the efficient handling of structured data in the form of DataFrame, which is useful for cleaning, transforming, and analyzing large datasets. It is Extremely reliable and extensively used in AI, machine learning, and data science applications.

## 4.3 The reliability of the Output file

The processed outputs are saved in TSV (Tab-Separated Values) format instead of CSV (Comma-Separated Values) for several reasons related to data structure, readability, and system efficiency. The basic structure of TSV is unlikely to change.

## 4.4 The reliability of the data platform

In this research, the **Databricks platform** will be used with dedicated clusters assigned for data analytics purposes, ensuring high computational power and reliability for processing and analyzing large datasets. The platform's distributed architecture provides fault tolerance and scalability, making it ideal for handling the operational data of mainframe systems.

As the research progresses into its steady state, the author intends to expand the solution's reliability and accessibility by integrating with AWS S3, Athena, and Power BI. This combination of

technologies allows for robust data storage, efficient query execution, and dynamic reporting capabilities, creating a more reliable licensed platform supported by reputable service providers.

#### **4.5 The reliability of the results**

The author has conducted thorough testing to validate the reliability of results by comparing manually processed outputs with results processed by advanced analytics. This included incorporating data pollution, to test the robustness of the solution, particularly focusing on dictionary lookup and generating alerts for new message IDs when they occurred. These tests ensured the detection and analysis processes functioned accurately under varied conditions. The developed artifact successfully reduced manual syslog processing time by 50% to 60% in real-time.

By manual comparison and by using the thesis research techniques, findings indicate that the results were reliable and trustworthy, forming a strong foundation for further behavioural analysis of system logs and operational trends. The testing methodology ensures that the analytics-based techniques align with the expected standards of accuracy and dependability.

### **5 About zSystem - Mainframe**

Mainframes are often referred to as **zSystems**, particularly in the context of IBM's mainframe line. Mainframe Servers continue to play a crucial role in various industries, with organizations relying on them for their reliability, scalability, and security in handling large-scale transactions (IBM, n.d.-c). For a more comprehensive understanding of mainframe systems, including their construction and robustness, readers are encouraged to watch a 23-minute YouTube video titled *Why Do Mainframes Still Exist? What's Inside One? 40TB, 200+ Cores, AI, and more!* (Dave's Garage, 2023) that offers an in-depth overview of the mainframe's architecture and its enduring capabilities. This video provides valuable insights into the evolution of mainframes and their critical role in modern computing.

As outlined Mainframe computers are distinct from other computing systems due to their capacity to handle vast amounts of data, provide high reliability, and support multiple users simultaneously, which are crucial for large-scale business operations (ZMainframes, 2024).

## 5.1 About Mainframe z facts

Mainframes, particularly IBM's zSeries systems, have evolved significantly over the past 60 years, yet remain indispensable due to their unmatched reliability, scalability, and adaptability. The latest IBM z16, introduced in April 2022, exemplifies this evolution with its incredible uptime performance of **99.999999%** ("nine nines"), a benchmark for resilience and reliability in mission-critical applications. Interesting Facts About z Mainframes (TechChannel, n.d.). as follows.

### IBM Z Fast Facts

- IBM Z platform experienced 3x growth in the last decade.
- The IBM Z platform supports 67 of the Fortune 100 firms.
- The IBM Z mainframe is used in 45 out of 50 of the world's top banks.
- IBM Z is in eight of the top 10 insurance companies.
- IBM Z is deployed by four out of the five top airlines.
- IBM Z is used by seven out of 10 top global retailers.
- IBM Z is in use by eight of the top 10 telecommunications providers.
- 24 of the top 25 countries by GDP have an IBM Z mainframe.
- Currently, more than 1,000 ISVs and open-source solutions are now on the platforms supporting every part of their product and services stacks.

In addition, ITIC 2023 Global Server Hardware and Server OS Reliability Report highlights the unparalleled reliability of mainframe systems, emphasizing their consistent uptime performance in mission-critical environments (Astecno, 2023)

According to the NASCIO-Accenture Cloud Study (2021), **89% of state CIOs** report that they rely on mainframes in some capacity, underlining the enduring importance of these legacy systems in modern IT infrastructures. The report highlights how government agencies continue to depend on mainframe computing for tasks such as data management, transaction processing, and ensuring operational continuity.

## 5.2 Db2 for z/OS architecture

In the context of this thesis report, while a comprehensive understanding of the entire Db2 for z/OS architecture may not be essential, it is beneficial to grasp the concept of Db2 address spaces. The zManav tool primarily gathers and compiles log statistics from the Master Task ssnmSTR (Subsystem Identifier Number MASTER TASK) of Db2 (IBM, n.d.-d), which is responsible for controlling local internal functions.

According to IBM (2023), Db2 functions as a formal subsystem within the z/OS environment, efficiently interacting with other z/OS subsystems and components. It operates as a distinct relational Database Management System (DBMS) instance, managing the creation, organization, modification, and access to database data. Db2 for z/OS processes execute in several z/OS address spaces, including a) the Database Services Address Space (ssnmDBM1), responsible for most database-related services; b) the System Services Address Space (ssnmMSTR), overseeing the starting, stopping, and local access control of Db2; and c) the Distributed Data Facility (DDF) Services Address Space (ssnmDIST), supporting network communications and remote database access requests. Additionally, Db2 uses d) the Internal Resource Lock Manager (IRLM) Address Space (irlmproc) for database resource control and locking, alongside other specialized address spaces for various functions.

## 5.3 About Mainframe Db2 System Programmers and responsibilities

A Mainframe Db2 Systems Programmer is a specialized IT professional responsible for managing and maintaining Db2 database systems on mainframe computers (IBM, n.d.-e). These individuals are experts in their field, possessing a deep understanding of both the Db2 database management system and the z/OS mainframe operating system. They often find themselves entangled in complex and challenging Business as Usual (BAU) tasks, along with the demanding responsibilities that come with system software and hardware upgrades. This role falls under the niche skill category due to the specific and advanced knowledge required to effectively manage mainframe resources. In an effort to simplify and streamline their daily monitoring tasks, a new tool named zManav is being introduced. zManav is designed to alleviate some of the burdens faced by Mainframe Db2 Systems Programmers by providing a simple graphical tool format on the required error logs and a trend on the system message ID and messages. This tool aims to ensure that the Db2 subsystems

health is maintained, facilitating the smooth handling of production workloads. zManav stands out as a supportive solution, enabling these professionals to focus more on strategic tasks while confidently managing the operational aspects of their Db2 environments.

## 5.4 BAU vs Major Upgrade challenges

When comparing Business As Usual (BAU) challenges with those of major upgrades in the context of Mainframe Db2 Systems Programming, distinct differences emerge. BAU challenges primarily involve ongoing maintenance and daily operational tasks. These include performance tuning, ensuring data integrity and security, handling resource constraints, and managing routine backups. BAU tasks require constant vigilance to maintain system stability and efficiency in the face of regular workload demands. On the other hand, major upgrade challenges are more episodic but significantly more complex. They encompass compatibility assessments, meticulous planning for minimal business disruption, rigorous testing of new features, data migration, and ensuring the integrity of the upgraded system. These upgrades demand extensive preparation and often involve navigating through uncharted issues that arise with new system versions. While BAU tasks are about maintaining and optimizing the current system state, major upgrades are transformative processes that evolve the system to a new state, introducing both opportunities and challenges in ensuring system reliability and performance. Both BAU and major upgrade challenges require a high level of expertise, but major upgrades typically demand more intensive resource allocation and strategic planning to successfully transition to new system capabilities.

## 5.5 Db2 sysprog BAU challenges

As a Db2 for z/OS Systems Programmer, there are several Business as Usual (BAU) challenges that we face. These challenges require a mix of technical expertise, problem-solving skills, and effective communication. Some of the common responsibilities and challenges include (IBM, n.d.-f):

1. **Performance Optimization:** Continuously monitoring and tuning the Db2 database for optimal performance can be challenging. This involves analyzing query performances, optimizing indexes, and balancing system resources.
2. **Managing Resource Constraints:** Dealing with limitations in terms of CPU, memory, or storage on the mainframe environment. Efficiently managing these resources to ensure smooth operations is a key challenge.

3. **Ensuring High Availability and Disaster Recovery:** Maintaining high availability of the Db2 databases and ensuring robust disaster recovery plans are in place. This includes dealing with unexpected outages and ensuring minimal downtime.
4. **Data Integrity and Security:** Ensuring the integrity and security of data in the Db2 environment. This involves implementing and enforcing security policies, regular auditing, and staying compliant with data protection regulations.
5. **Keeping Up with Patches and Upgrades:** Regularly applying patches - Recommended Service Upgrade, RSUyymm and staying current with Db2 upgrades can be challenging, especially ensuring compatibility with existing applications and minimizing downtime during the upgrade process.
6. **Capacity Planning:** Anticipating future growth and requirements in terms of database storage, processing power, and memory needs. This requires accurate forecasting and proactive planning.
7. **Troubleshooting and Problem Resolution:** Quickly diagnosing and resolving issues that arise in the Db2 environment. This often requires in-depth knowledge of Db2 internals and the ability to work under pressure.
8. **Collaboration with Other Teams:** Effectively communicating and collaborating with application development teams, network engineers, and other IT staff. This includes understanding their needs and ensuring that the database environment aligns with the overall IT strategy.
9. **Cost Management:** Balancing performance and resource needs with budget constraints. This includes optimizing resource usage and making cost-effective decisions.
10. **Staying Informed of New Features and Trends:** Keeping up to date with the latest developments in Db2 and the broader technology landscape, and understanding how these can be applied to improve the existing environment.

These challenges require a Db2 Systems Programmer to be adaptable, continually learning, and proficient in problem-solving to ensure the smooth operation of Db2 databases on z/OS systems.

## 5.6 Db2 sysprog Upgrade challenges

Alongside Mainframe hardware upgrade, a software upgrade like Db2 subsystem to a latest version in a z/OS environment, presents a number of challenges for a Systems Programmer. These challenges arise from the complexity of the system, the critical nature of the data involved, and the need for minimal disruption during the upgrade process. Here are some of the key challenges (Sbai Idrissi, 2023):

1. **Compatibility Issues:** Ensuring that the new version of Db2 is compatible with existing applications and software. This includes verifying that all connected applications, tools, and utilities will function correctly with the new Db2 version.
2. **Planning and Scheduling:** Coordinating the timing of the upgrade to minimize impact on business operations. This often involves planning the upgrade for off-peak hours and ensuring all stakeholders are informed.

3. **Data Migration and Integrity:** Safeguarding the integrity of data during the upgrade process. It's crucial to ensure that all data is accurately migrated and that there are no data losses or corruption.
4. **Testing:** Rigorously testing the new system before going live to ensure all components work as expected. This includes performance testing, regression testing, and disaster recovery testing.
5. **Downtime Management:** Reducing downtime is a major challenge. Even with careful planning, there's often a need for some system downtime during the upgrade, which has to be managed efficiently.
6. **Backup and Recovery Procedures:** Ensuring robust backup and recovery procedures are in place before starting the upgrade. This is essential to recover data in case something goes wrong during the upgrade.
7. **Training and Documentation:** Updating documentation and training staff on new features and changes in the Db2 environment. This is important for the smooth operation and maintenance of the system post-upgrade.
8. **Resource Allocation:** Ensuring adequate resources are available for the upgrade process, including necessary hardware and software resources, as well as human resources for implementation and troubleshooting.
9. **Performance Tuning:** After the upgrade, there may be a need for performance tuning to optimize the new system's settings and configurations for the best performance.
10. **Compliance and Security:** Ensuring the upgraded system meets all regulatory compliance requirements and that new security features are configured and tested.

Handling these challenges requires meticulous planning, thorough testing, effective communication, and a deep understanding of both Db2 and the z/OS environment.

## 6 Thesis Design and development

Considering the fact that Mainframe subsystems log/ syslogs are typically in a format that is not easily readable by data analytics tools. By converting them to TSV (tab-separated values) files, we can transform the data into a structured format that is easier to analyze. TSV files can be easily imported into spreadsheets or data analysis tools, allowing us to perform more advanced analysis on the data. Additionally, TSV files can be compressed to reduce storage space, making it easier to store and manage large amounts of data.

In this research, I will meticulously collect Db2 Master Task (JESMSGSLG from MSTR) logs on a daily basis to perform an in-depth analysis of the Db2 system's health and behaviour. This process involves extracting relevant Db2 JESMSGSLG from Mainframe using REXX language program as TEXT file and transfer to AWS S3 bucket to load them into Databricks as TXT file and convert to TSV

(Tab Separated Value) file using Pandas Data frame Data Analytics , aligning with the data analytics lifecycle's emphasis on proper data gathering and preparation (Northeastern University, 2020). Subsequently, I plan to use aws Athena for storing the metrics from analytics and populate the results using Microsoft Power Bi, a licensed interactive data visualization tool for monitoring, to visualize and interpret this data. The initial process is to handle only within databricks on small scale data. Through this integration, I aim to create intuitive dashboards that will reveal critical insights into the Db2 system, such as resource utilization, error frequency, and performance trends, thereby facilitating proactive system maintenance and optimization.

The converted TSV files are further used by for variety of data analytics techniques to gain more insights and make data-driven decisions such as a) Trend analysis; b) Anomaly detection; c) Predictive modeling by using machine learning algorithms, we can build models that can predict future events or behaviours based on historical data.

It is also intended to maintain the historical data by storing and reusing processed TSV file to predict overall system behaviour trends on demand.

Refer (Figure 2) for the High-Level design explain the overall process with data transformation.

In the future expansion, the manual FTP process will be made automated through batch job to FTP the required contents from Mainframe to AWS S3 bucket.



Figure 2 Research Design

## 6.1 Thesis Research strategy

The approval process of securing and setup the AWS and Power BI platform for end to end monitoring solution is still under progress within organization (Refer Table 1). Hence author focuses

only with key technique between Mainframe and Databricks addressing the Research Questions 1 (RQ1) as POC (Proof of Concept).

Table 1: Platform Readiness

Platform	Purpose	Status	Comments
Mainframe Batch	Mainframe automated batch JCL (Job Control Language) job to ftp the desired syslogs to target servers.	To be developed	Right now, managing manually.
aws S3 (Simple Storage Service)	To save the syslogs in TEXT format which are FTP transferred from Mainframe.	To be secured	Using my laptop as a medium, right now managing the FTP txt file in Databricks for thesis research purpose.
Databricks	Data Analytics in secured and reliable platform	Available	Databricks is where the entire research is carried out for the thesis .
aws Athena	A serverless, interactive query service provided by aws that allows you to analyze data stored in Amazon S3 .To use SQL syntax to query struc-	To be secured	

	tured, semi-structured, or unstructured data that are developed in the research		
Power BI	Power BI offers a wide range of visualizations (charts, graphs, maps, tables) to display data in an intuitive and interactive way. Develop query based charts .	To be secured	

STEP 1: Identify your source Db2 subsystem JESMSGLOGS also known as Job Entry Subsystem Message Log. This dataset records system and JES messages generated during job execution. It includes messages about job processing, routing, scheduling, and any errors detected by JES. According to IBM (IBM, n.d.-g), the Job Entry Subsystem (JES) plays a critical role in managing jobs within the mainframe environment (Job entry subsystem 2/3, 2024). These logs are extracted using REXX (Restructured Extended Executor) is a versatile scripting language designed for use in IBM mainframes (IBM Documentation, n.d.-a).

Refer (Figure 3) for a snippet of the REXX code used to print the logs into a flat file. The key point here is the number of lines you want to print. By scrolling JESMSLG to the end of page we can get this information.

```

QUEUE "PRE DB2MSTR" /* SDSF COMMANDS IN BATCH */
QUEUE "ST"
QUEUE "SEL DDNAME JESMSG LG" /* SELECT ONLY JESMSG LG DD */
QUEUE "FILTER JOBNAME " JOBID
QUEUE "++S"
QUEUE "XDC"
QUEUE "SE 'JESMSG LG'" /* SELECT ONLY JESMSG LG */
QUEUE "PRINT FILE TEMPPRT" /* PRINT TO TEMP DATASET */
QUEUE "PRINT 1 99999999"
QUEUE "PRINT CLOSE"
QUEUE "END"
QUEUE "EXIT"
"EXECIO" QUEUED() "DISKW ISFIN (FINIS" /* INPUT TO SDSF BATCH */
ADDRESS ISPEXEC "SELECT PGM(ISFAFD) PARM('++25,80') " /* INVOKE SDSF */
EXIT
***** Bottom of Data *****

```

Figure 3 REXX JESMSG LG extraction

STEP 2: Transfer the source file where SYSLOGS are saved to author's laptop in text/ASCII format from Mainframe. For security reasons, ID name and file names are masked. Refer (Figure 4) for a FTP selections.

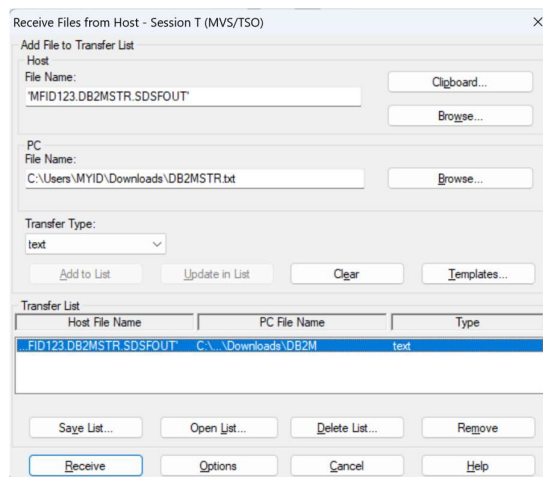


Figure 4 FTP selection in Mainframe

STEP 3: Import the file to Databricks in target folder

Step 4: Data format conversion from TEXT to TSV file. As stated above, Mainframe subsystems log/syslogs are typically in a format that is not easily readable by data analytics tools. By converting them to TSV (tab-separated values) files, we can transform the data into a structured format that is easier to analyze. I used Expressions and Pandas for initial data manipulation and analytics for processing the input file. Refer (Figure 5) This section of the code merges continuation lines into single entries. It checks for continuation lines by looking for lines that start with a number and have extra content. If a line is a continuation line, it is merged with the previous line in the buffer. If a line ends with a number, it is treated as the start of a new log entry. Regular lines or standalone lines without continuation are added to the log entries list. This is one of the key requirements in the alignment.

```

log_entries = []
buffer = ''

for line in syslog_lines:
    line_stripped = line.strip()

    continuation_match = re.match(r'^\s*(\d+)\s+(.*)', line_stripped)

    if continuation_match and buffer:
        continuation_message = continuation_match.group(2)
        buffer = re.sub(r'^\s*(\d+)\s+', '', buffer)
        buffer += ' ' + continuation_message.replace(continuation_match.group(1), '', 1)

    elif re.search(r'(\d+)\s*$', line_stripped):
        if buffer:
            log_entries.append(buffer)
            buffer = line_stripped.replace(re.search(r'\d+\s*$', line_stripped).group(), '', 1)
        else:
            if buffer:
                log_entries.append(buffer)
                buffer = ''
            log_entries.append(line_stripped)

    if buffer:
        buffer = re.sub(r'^\s*(\d+)\s+', '', buffer)
        log_entries.append(buffer.strip())

```

Figure 5 Merge Continuation Lines

Step 5: Parse each line of the log file and extract structured columns for further analysis. It uses regular expressions to extract individual fields from each log entry, including the timestamp, STCID, Db2msgid, subsystem ID, and log message. This one other key requirement in the research. Refer (Figure 6) It also extracts the date from the log entry string using regex. If the Db2msgid value is not in the exclusion list, it is added to the parsed\_data list and its date and timestamp are added to the db2\_msgid\_dates dictionary. If a message ID is in the exclusion list, the line is added to the excluded syslog list. Overall, this code keeps track of the dates and timestamps associated with each db2\_msg\_id seen in the log file. It also keeps track of new db2\_msg\_id values that were not seen before. This is a breakthrough in addressing the research question RQ2.

```

parsed_data = []
excluded_syslog = []
current_date = None
db2_msgid_dates = {}
new_db2_msgids = set()

for line in log_entries:
    match = re.match(r'^(\d{2})\.\d{2})\.\d{2})\s+(\S+)\s+(\S+)\s+(\S+)\s+(\S+)\s+(.*)', line.strip())

    if match:
        timestamp = match.group(1)
        stcid = match.group(2)
        db2_msg_id = match.group(3)
        subsystem_id = match.group(4)
        message = match.group(5)

        date_match = re.search(r'\b(\d{2})\s+(\d{4})\b', line)
        if date_match:
            log_date = pd.to_datetime(date_match.group(1), format='%d %b %Y').date()
        else:
            log_date = current_date

        if current_date is None or log_date > current_date:
            current_date = log_date

        if db2_msg_id not in ["IGD01007I", "IGD01008I", "IGD01009I", "IGD01010I", "IGD17364I", "IGD641I"]:
            parsed_data.append([log_date, timestamp, stcid, db2_msg_id, subsystem_id, message])

            if db2_msg_id not in db2_msgid_dates:
                db2_msgid_dates[db2_msg_id] = [(log_date, timestamp)]
                new_db2_msgids.add(db2_msg_id)
            else:
                db2_msgid_dates[db2_msg_id].append((log_date, timestamp))

        else:
            excluded_syslog.append(line.strip())

```

Figure 6 Parse to structured columns

Step 7: Convert the parsed\_data list into a pandas DataFrame with columns 'Date', 'Time', 'STCID', 'Db2msgid', 'SSID', and 'Message'. Generate an alert if new db2\_msgid values are detected. The exclusion list is used to exclude certain db2\_msgid values from the alert. We achieve in converting the parsed data into a panda DataFrame, saves the DataFrame to a TSV file, updates the db2\_msgid\_dates dictionary, and generates an alert if new db2\_msgid values are detected. Refer (Figure 7) on the formatted output in TSV file.

```

# Step 4: Convert the parsed data into a pandas DataFrame
df = pd.DataFrame(parsed_data, columns=['Date', 'Time', 'STCID', 'Db2msgid', 'SSID', 'Message'])
# Step 5: Save the DataFrame to a TSV file and update the Db2msgid dates dictionary
filename = '/Workspace/Users/AB7028@student.jamk.fi/JAMK_Thesis/T301NSTR.tsv'
df.to_csv(filename, sep='\t', index=False)
print(f"Output saved to '{filename}'.")

# Update the Db2msgid dates dictionary with any new dates and timestamps
for db2_msgid in new_db2_msgids:
    if db2_msgid not in db2_msgid_dates:
        db2_msgid_dates[db2_msgid] = []
        db2_msgid_dates[db2_msgid].extend([(date, time) for date, time in df.loc[df['Db2msgid'] == db2_msgid, ['Date', 'Time']].itertuples(index=False, name=None)])

# Step 6: Generate alert if new Db2msgid values are detected
if new_db2_msgids:
    print(f"New Db2msgid values detected: ('', '.join(new_db2_msgids)). Please investigate.")
else:
    print("No new Db2msgid values detected.")

# Step 6: Generate alert if new Db2msgid values are detected (excluding those in the exclusion list)
new_db2_msgids_excl = [msg_id for msg_id in new_db2_msgids if msg_id not in ["DSNR004I", "DSNL523I", "DSNA650I", "DSN3582I", "DSNI049I", "DSNR002I", "DSNB233I", "DSNZ016I", "DSNR006I",
"CAS9800I", "DSNG007I", "DSNJ099I", "IGD14I", "DSNV024I", "DSNV001I", "----", "DSNR003I", "DSNB317I", "DSNR001I", "DSNJ127I", "DSNJ139I", "DSNI500I", "DSNJ001I", "DSN3029I", "DSNL003I",
"DSNL512I", "DSNV434I", "DSNL511I", "DSNJ002I", "DSN3583I", "CAS9803I", "DSNZ017I", "DSNZ002I", "DSN3201I", "DSNJ003I", "DSN9022I", "IEF695I", "DSNR005I", "DSNJI", "DSNL510I", "IXL014I",
"DSNB260I", "AHASP373", "DSNL004I"]]
if new_db2_msgids_excl:
    print("New Db2msgid values detected. Please investigate.")
    for msg_id in new_db2_msgids_excl:
        message = df.loc[df['Db2msgid'] == msg_id, 'Message'].iloc[0]
        print(f" {msg_id}: {message}")
else:
    print("No new Db2msgid values detected.")

```

Figure 7 Save Dtaframe as TSV

## 7 Results : Thesis research outcome

Given the significance of mainframes, IBM provides exceptional documentation for their hardware and software. A comprehensive repository of documentation exists for each version and release. This research focuses specifically on the Db2 V13 message ID reference (*IBM Documentation*, n.d.-b).

In the author's office under mainframe administration department, an old saying goes, '*If you cannot measure and monitor, you cannot manage.*' In line with this principle, we now have the necessary metrics for effective daily monitoring. However, additional tweaks are needed to refine the overall research and implementation. The author is committed to addressing these enhancements to ensure the continuous development and improvement of the zManav's capabilities.

### 7.1 DSNJ002I Full Active Log

We will handpick few scenarios to get a trend of daily Active LOGS from DSNJ002I (*IBM Documentation*, n.d.-c). In this given case, Db2 SUBSYSTEM was started on TUESDAY, 05 NOV 2024 and active up and running. In general Mainframe servers are not IPLed (Initial Program Load-Restart of services) unless it is. In many Mainframe shops, Mainframe services are maintained 24X7 for many months or even for years due to business-critical applications. So, the JESMSGLG from started

tasks is active and appends the daily activities throughout their life cycle. Refer (Figure 8). This addresses Research question (RQ3).

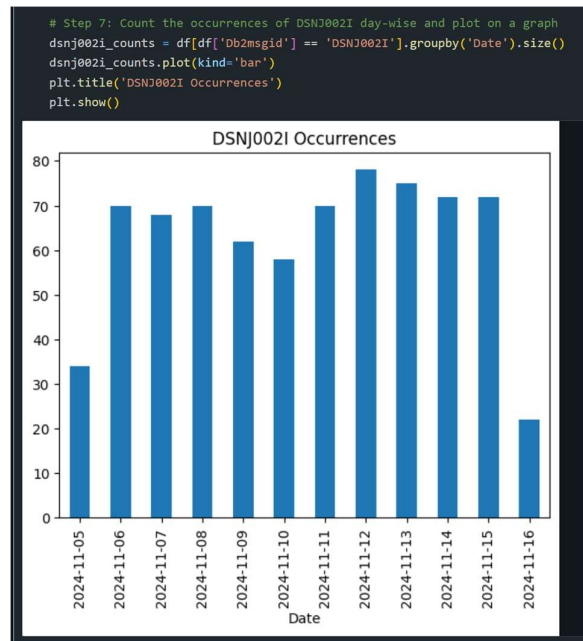


Figure 8 DSNJ002I Full Active log

## 7.2 DSNJ139I OFFLOAD TASK ENDED

Refer (Figure 9) for the DSNJ139I (*IBM Documentation, n.d.-d*) where we get the daily metrics of ACTIVE LOGS OFFLOADED TO ARCHIVE LOGS.

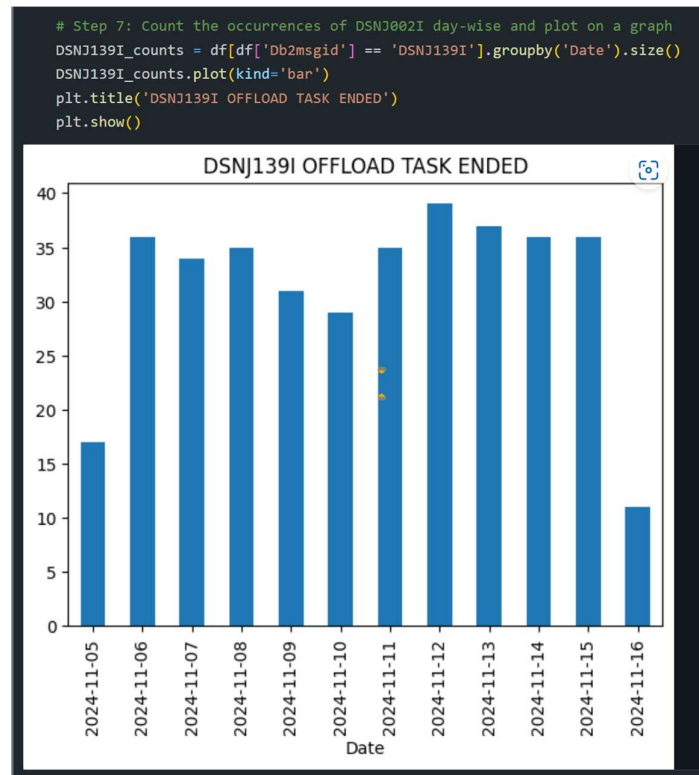


Figure 9 DSNJ139I LOG OFFLOAD TASK ENDED

### 7.3 DSNQ022I MQ listener Failed to connect.

The input file was intentionally polluted with a new message ID to trap and identify the alert using the DataFrame dictionary. Refer (Figure 10)

```
New Db2msgid values detected: DSNQ022I. Please investigate.
```

Figure 10 DSNQ002I MQ LISTNER FAILED

## 8 Developments

The development of this thesis involved a systematic approach to exploring the research problem, gathering relevant data, and analyzing findings to address the identified gaps.

Methodologies were carefully designed to ensure the collection of accurate and meaningful data, followed by rigorous analysis using appropriate techniques.

The culmination of these efforts is a cohesive body of work that contributes to both theoretical understanding and practical applications in the thesis research.

## **8.1 Tool as user interactive**

Given the rising workload and the increased activity of JESMSG LG, the growth footprint of JESMSG LG has been significant. As workloads increase—whether through additional queries, transactions, or batch jobs—the system generates a higher volume of messages.

These include:

- Database operation logs, archiving details, and resource contention alerts.
- Error and informational messages, especially when workload leads to contention or systemic issues.

To address this, the author decided to introduce user interactive selection process TEST and DEVELOPMENT Db2 systems. This even simplified the trending process.

The results were quite impressive, highlighting opportunities for further improvement.

Leveraging below data analytics techniques, it was achieved. Refer (Figure 11)

```

import pandas as pd
import matplotlib.pyplot as plt

# Define a function to get the top 10 values of Db2msgid that have counts greater than 250
def get_top_10(db2_file):
    # Read the TSV file into a pandas DataFrame
    df = pd.read_csv(db2_file, sep='\t')

    # Get the count of each Db2msgid value
    db2_msgid_counts = df.groupby('Db2msgid').size().reset_index(name='Count')

    # Filter the top 10 values with counts greater than 250
    top_10 = db2_msgid_counts[db2_msgid_counts['Count'] > 250].sort_values(by='Count', ascending=False).head(10)

    # Return the top 10 values as a list of tuples
    return [(row['Db2msgid'], row['Count']) for _, row in top_10.iterrows()]

# Define a function to generate a graph for a given Db2msgid value
def generate_graph(db2_file, db2_msgid):
    # Read the TSV file into a pandas DataFrame
    df = pd.read_csv(db2_file, sep='\t')

    # Filter the DataFrame to include only rows with the given Db2msgid value
    data = df[df['Db2msgid'] == db2_msgid]

    # Group the data by date and count the occurrences of the Db2msgid value
    counts = data.groupby('Date').size()

    # Create a bar chart using matplotlib
    counts.plot(kind='bar')
    plt.title(f"{db2_msgid} Occurrences")
    plt.show()

# Create a dropdown menu to select a DB2 environment (TEST or DEVELOPMENT)
print("Please select a DB2 environment:")
print("1. TEST")
print("2. DEVELOPMENT")

# Get the user's selection
while True:
    try:
        db2_env = int(input("Enter the number of the DB2 environment: "))
        if db2_env in [1, 2]:
            break
        else:
            print("Invalid selection. Please try again.")
    except ValueError:
        print("Invalid input. Please enter a number.")

# Set the DB2 file based on the user's selection
if db2_env == 1:
    db2_file = '/Workspace/Users/AB70282@student.jamk.fi/JAMK_Thesis/TEST.tsv'
else:
    db2_file = '/Workspace/Users/AB70282@student.jamk.fi/JAMK_Thesis/DEVELOPMENT.tsv'

# Get the top 10 values of Db2msgid that have counts greater than 250
top_10 = get_top_10(db2_file)

# Loop until the user selects "0" to exit the program
while True:
    # Create a dropdown menu to select a Db2msgid value (or enter 0 to exit)
    print("\nPlease select a Db2msgid value (or enter 0 to exit):")
    for i, (db2_msgid, count) in enumerate(top_10):
        print(f"{i+1}. {db2_msgid} ({count} occurrences)")
    print("0. Exit")

    # Get the user's selection
    while True:
        try:
            selection = int(input("Enter the number of the Db2msgid value: "))
            if 0 <= selection <= len(top_10):
                break
            else:
                print("Invalid selection. Please try again.")
        except ValueError:
            print("Invalid input. Please enter a number.")

    # Generate a graph for the selected Db2msgid value (if not exiting)
    if selection == 0:
        break
    else:
        db2_msgid = top_10[selection-1][0]
        generate_graph(db2_file, db2_msgid)

```

Figure 11: User Interactive technique

## 8.2 Focused Trend Analysis approach using user interactive options

As part of this process, the author chose to focus exclusively on message IDs that appeared more than 250 times during the selection process. This threshold was set to ensure attention was directed toward high-frequency occurrences, which are more likely to reflect significant trends or systemic behaviours.

By narrowing the scope:

1. **Efficient Trend Identification:** Reduced noise by excluding low-frequency or isolated events, making the analysis more streamlined and impactful.
2. **Actionable Insights:** Focused on recurring issues or patterns that could indicate bottlenecks, contention points, or system inefficiencies.
3. **Data-Driven Decisions:** Leveraged these frequently occurring message IDs to prioritize areas for system improvement and optimization.

This focused methodology provided a clear path to identifying trends and addressing underlying causes effectively.

Below is an overview, refer (Figure 12) of the selection process in **TEST** and **DEVELOPMENT** environments. A user-friendly, interactive interface was provided to enable users to filter and select specific messages based on their preferences. Refer (Figure 13).

```
Please select a DB2 environment:
1. TEST
2. DEVELOPMENT

Enter the number of the DB2 environment: 2
```

Figure 12 User Interactive Db2 Selection

Below are some notable message IDs that stood out:

```
Please select a DB2 environment:
1. TEST
2. DEVELOPMENT

Enter the number of the DB2 environment: 2

Please select a Db2msgid value (or enter 0 to exit):
1. DSNJ031I (7541 occurrences)
2. DSNB260I (3231 occurrences)
3. DSNJ001I (2043 occurrences)
4. DSNJ002I (2041 occurrences)
5. DSNJ003I (2040 occurrences)
6. DSNL511I (1124 occurrences)
7. DSNJ139I (1022 occurrences)
8. DSNR035I (709 occurrences)
9. DSN3201I (658 occurrences)
0. Exit

Enter the number of the Db2msgid value: 2
```

Figure 13 User Interactive Message ID Selection

**8.2.1 DSNJ031I WARNING - UNCOMMITTED UR HAS WRITTEN number LOG RECORDS.**

As anticipated by readers, the author chose to analyze the trend of DSNJ031I out of curiosity. The results were striking, clearly indicating ample scope for further investigation into the issue. Refer (Figure 14) for a comprehensive view of this trend. The pattern observed is highly unusual and warrants further attention. DSNJ031I indicates that a transaction is generating an unusually high volume of logging activity. The transaction is acquiring a large number of local locks, which might not be propagated effectively. This can lead to global lock contention, where multiple systems or workloads compete for shared resources, degrading performance.

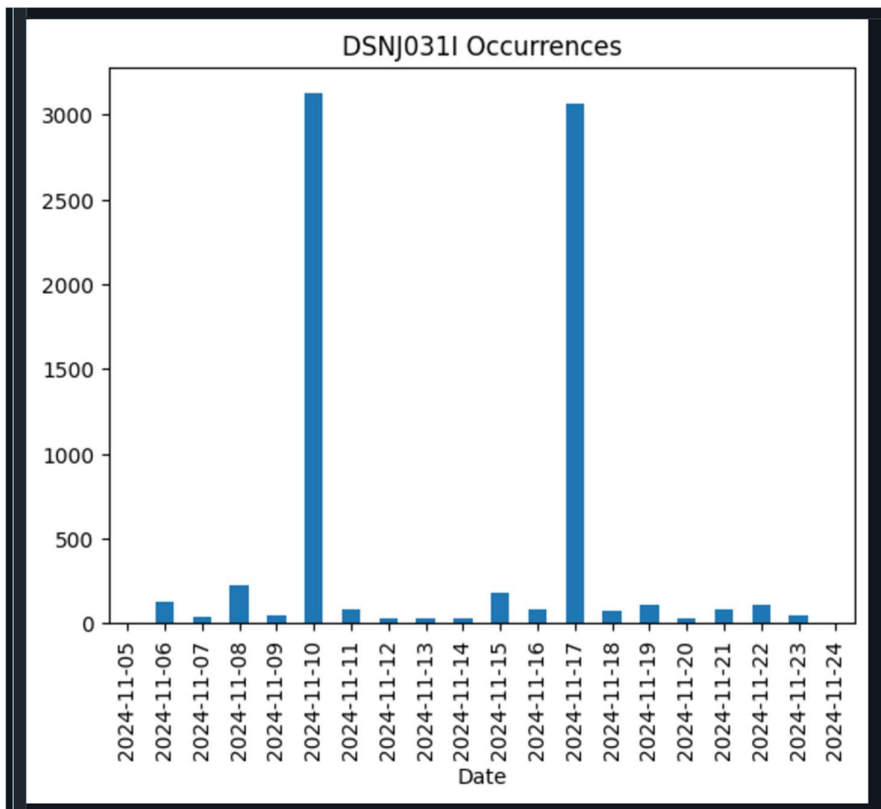


Figure 14 DSNJ031I WARNING -UNCOMMITTED UR HAS WRITTEN number LOG RECORDS

### 8.2.2 DSNB260I WARNING - A READER HAS BEEN RUNNING FOR duration MINUTES.

The DSNB260I, are sometimes treated as runaway threads, that are from poorly optimized queries, lock contentions, loops, waiting for I/O. In this case, refer (Figure 15) trend. It was found to be application specific intensive testing conducted on 8<sup>th</sup> November has resulted such a high spike. Further investigations can be carried out with associated application team to analyse the SQL or stored procedure and optimize better.

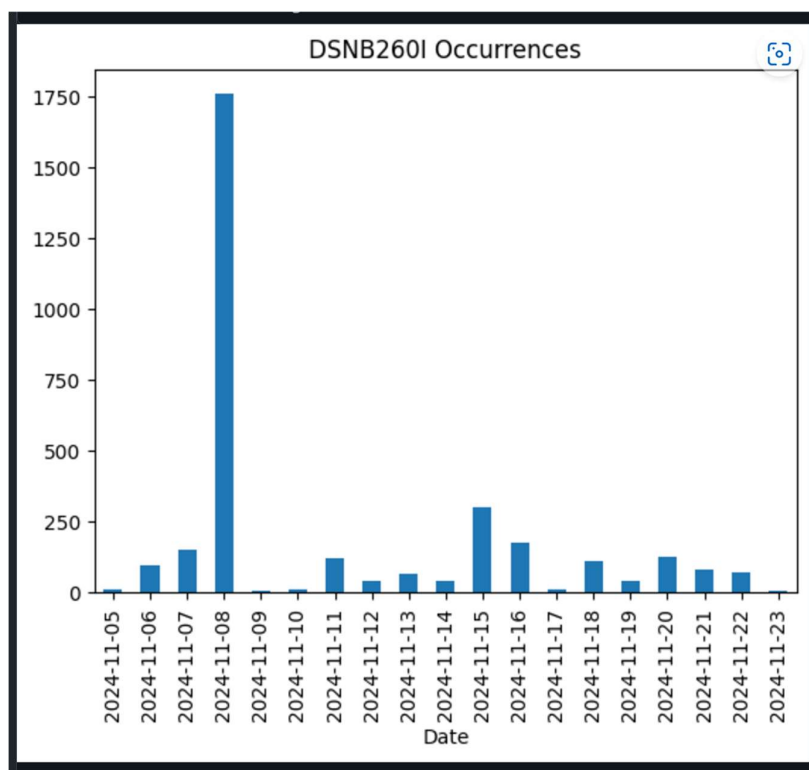


Figure 15 DSNB260I WARNING - A READER HAS BEEN RUNNING FOR duration MINUTES

### 8.2.3 Focused Trend Analysis approach on Db2 for z/OSGroup Buffer Pool Shortages

Using zManav capabilities, While author started with new Db2 subsystem Master log analysis, the new message ID technique revealed a potential alerts related to Db2 group Buffer Pool's.

If readers wish to know more about Group buffer Pool, they must understand Db2 for z/os high availability architecture, ability to operate in a data-sharing group, where multiple Db2 subsystems (members) share access to the same database, enabling workload balancing and failover capabilities. This is achieved through Parallel Sysplex, a clustering technology in z/OS that uses a Coupling Facility (CF) to manage shared resources like Group Buffer Pools, lock structures, and shared cache.

In a nutshell, group buffer pool is a critical component of Db2 High availability setup . These Cache structure are defined as per requirement (IBM, n.d.-h; IBM, n.d.-i ).

Where GBPxxx is the name of the group buffer pool; the following restrictions apply:

- 4 KB group buffer pools are named GBP0, GBP1, ..., GBP49
- 8 KB group buffer pools are named GBP8K0, GBP8K1, ..., GBP8K9
- 16 KB group buffer pools are named GBP16K0, GBP16K1, ..., GBP16K9
- 32 KB group buffer pools are named GBP32K, GBP32K1, ..., GBP32K9

zManav detected three main alerts as new message ID while new logs were processed.

\*DSNB319A DSNB1CNE THERE IS A SHORTAGE OF SPACE

\*DSNB325A DSNB1CNE THERE IS A CRITICAL SHORTAGE

DSNB327I DSNB1CNE GROUP BUFFER POOL HAS ADEQUATE FREE SPACE

Refer (Figure 16) on the data analytics technique used to extract the sum total of these system critical alerts on Group Buffer Pool.

```

import pandas as pd

# Step 1: Read the TSV file into a pandas DataFrame
df = pd.read_csv('/Workspace/Users/AB7028@student.jamk.fi/JAMK_Thesis/D301MSTR_2.tsv', sep='\t')

# Step 2: Extract the bufferpool name from the entire row, or return NaN if no match is found on the Db2msgid is not one of the desired values
df['Bufferpool'] = df.apply(lambda row: row.astype(str).str.extract('(GBP\d+-?\w*)', expand=False).dropna().iloc[0] if row['Db2msgid'] in ['*DSNB319A', '*DSNB325A', '*DSNB327I'] and row.
astype(str).str.extract('(GBP\d+-?\w*)', expand=False).dropna().shape[0] > 0 else pd.NaT, axis=1)

# Step 3: Filter the DataFrame to include only rows with the desired message IDs
messages = df[df['Db2msgid'].isin(['*DSNB319A', '*DSNB325A', '*DSNB327I'])]

# Step 4: Filter the DataFrame to include only rows with GBP bufferpools
messages = messages[messages['Bufferpool'].str.startswith('GBP')]

# Step 5: Group the data by datetime, bufferpool, message ID, and count the occurrences
counts = messages.groupby(['Bufferpool']).size()

# Step 6: Print the sum total of each GBP count
print("Sum total of each GBP count:")
print(counts)

Sum total of each GBP count:
Bufferpool
GBP0      2
GBP0-SEC  2
GBP4      6
GBP4-SEC  6
GBP5     24
GBP5-SEC  24
dtype: int64

```

Figure 16 Group Buffer Alert Pool Sum Total

Refer (Figure 17) on the data analytics techniques used to extract trend of each Group buffer Pool that required system programmer further attention.

```

import pandas as pd
import matplotlib.pyplot as plt

# Step 1: Read the TSV file into a pandas DataFrame
df = pd.read_csv('/Workspace/Users/AB7028@student.jamk.fi/JAMK_Thesis/D301MSTR_2.tsv', sep='\t')

# Step 2: Extract the bufferpool name from the entire row, or return "unknown" if no match is found
df['Bufferpool'] = df.apply(lambda row: row.astype(str).str.extract('(GBP\d+-?\w*)', expand=False).dropna().iloc[0] if row.astype(str).str.extract('(GBP\d+-?\w*)', expand=False).dropna().
shape[0] > 0 else "unknown", axis=1)

# Step 3: Filter the DataFrame to include only rows with the desired message IDs
red_messages = df[df['Db2msgid'] == '*DSNB319A'].copy()
red_messages['Color'] = 'lightred'
blue_messages = df[df['Db2msgid'] == '*DSNB325A'].copy()
blue_messages['Color'] = 'lightblue'
green_messages = df[df['Db2msgid'] == '*DSNB327I'].copy()
green_messages['Color'] = 'lightgreen'
messages = pd.concat([red_messages, blue_messages, green_messages])

# Step 4: Filter the DataFrame to include only rows with GBP bufferpools
messages = messages[messages['Bufferpool'].str.startswith('GBP')]

# Step 5: Group the data by datetime, bufferpool, message ID, and count the occurrences
counts = messages.groupby(['Date', 'Bufferpool', 'Db2msgid']).size().unstack().fillna(0)

# Step 6: Rename the columns to the desired names
counts = counts.rename(columns={'*DSNB319A': '*DSNB319A (SHORTAGE)', '*DSNB325A': '*DSNB325A (CRITICAL)', '*DSNB327I': '*DSNB327I (ADEQUATE)'})

# Step 7: Print the output with date and time information and save to a text file
print("Counts of *DSNB319A, *DSNB325A, and *DSNB327I grouped by GB and Datetime:")
print(counts)
counts.to_csv('/Workspace/Users/manigandan.ravichandran@op.fi/JAMK_Thesis/GBPSHORTAGE.txt', sep='\t')

# Step 8: Plot the trends for *DSNB319A, *DSNB325A, and *DSNB327I message IDs for each bufferpool
for bufferpool in counts.index.get_level_values('Bufferpool').unique().tolist():
    bp_data = counts.loc[slice(None), bufferpool, :]
    ax = bp_data.plot(kind='bar', stacked=True, color=['red', 'blue', 'green'])
    plt.title(f'{bufferpool}')
    ax.legend(loc='center left', bbox_to_anchor=(1.0, 0.5))
    plt.show()

```

Figure 17 Group BufferPool trend technique

### 8.2.4 Trend of DSNB319A (Shortage), DSNB325A (Critical), DSNB327I (Adequate)

The new Db2 logs that were in scope are from 5<sup>th</sup> November 24<sup>th</sup> November . It was quite interesting and also surprising to notice there was a chocking issue with Group Buffer Pool. This can be further considered for Mainframe Optimization and tuning requirements.

Refer (Figure 18,19,20,21,22,23) on the trend extracted for each Group buffer Pool alerts.

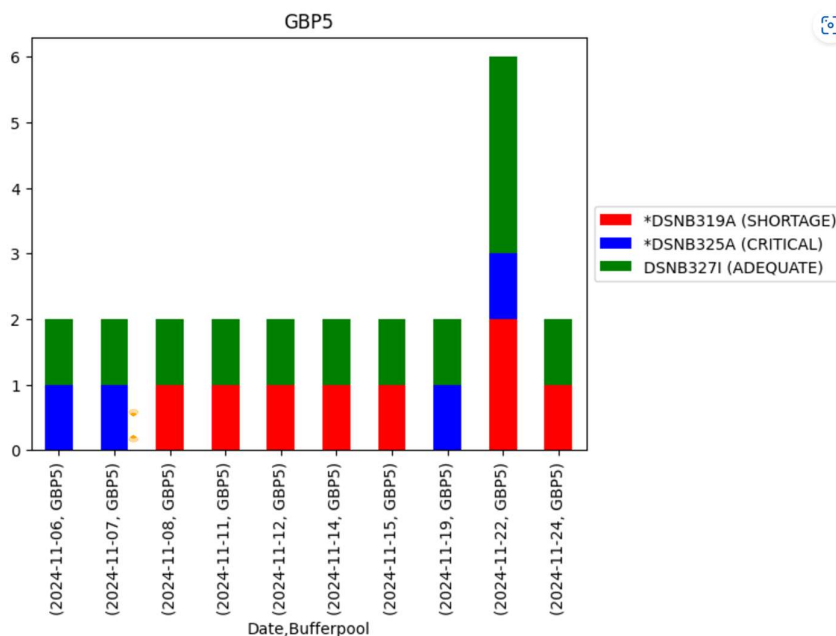


Figure 18 GBP5 TREND

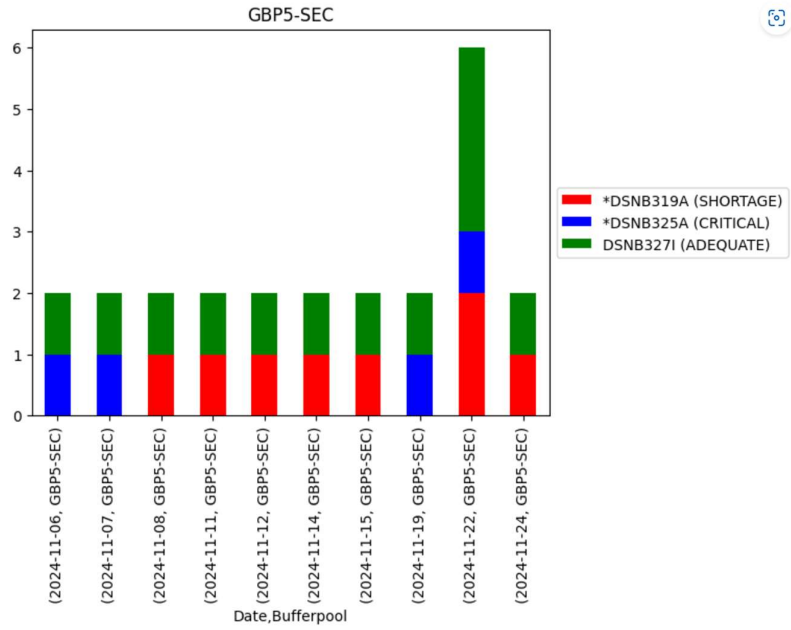


Figure 19 GBP5-SEC TREND

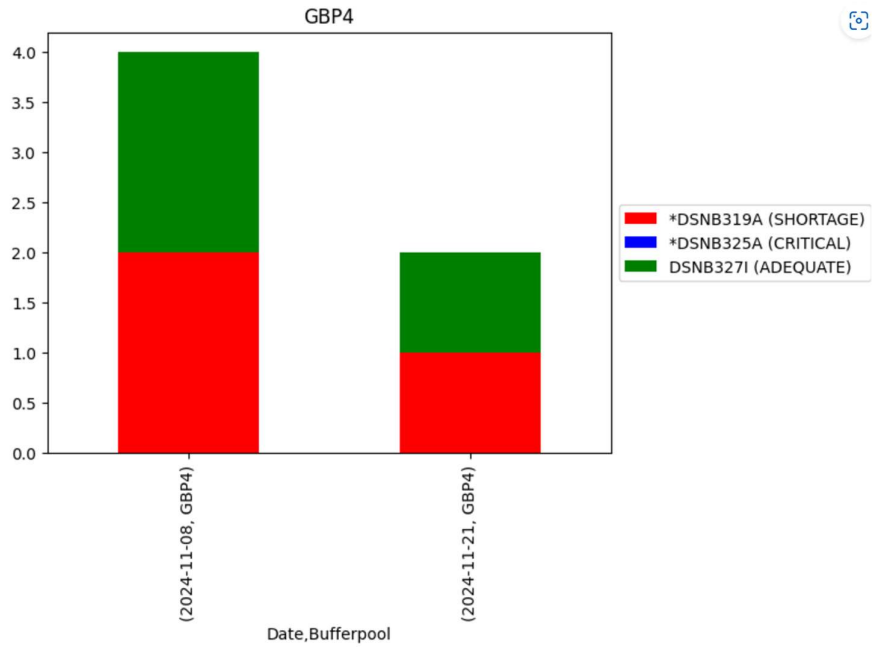


Figure 20 GBP4 TREND

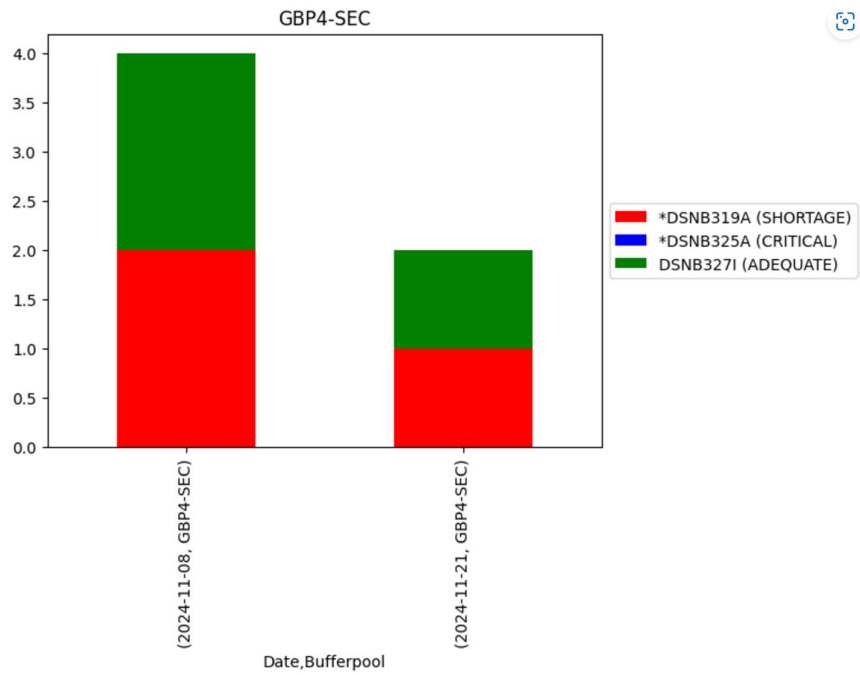


Figure 21 GBP4-SEC TREND

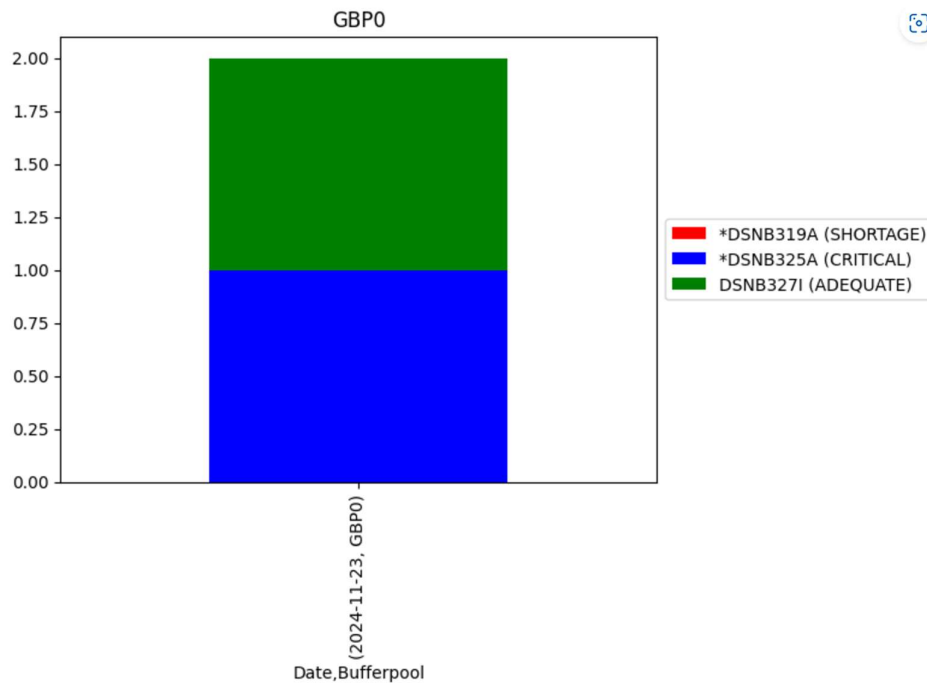


Figure 22 GBP0 TREND

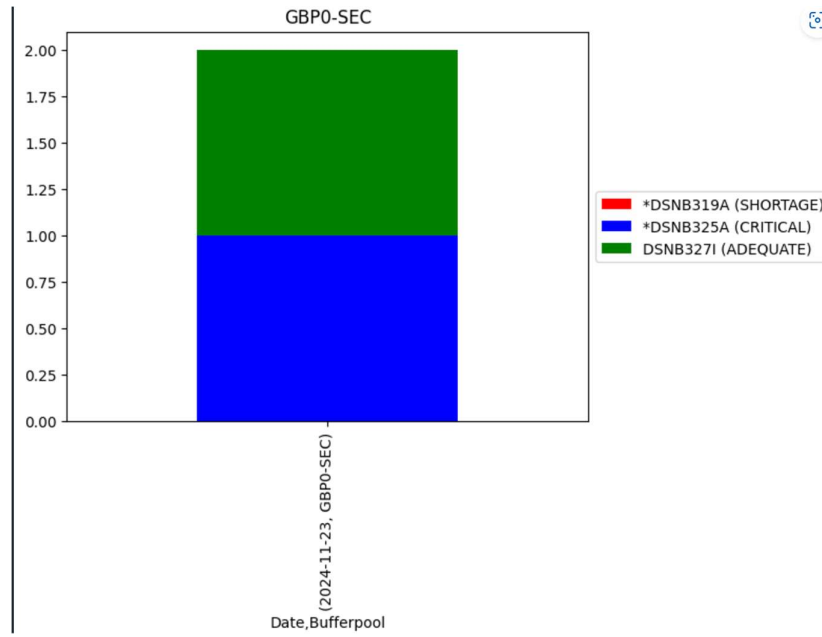


Figure 23 GBP0-SEC TREND

By achieving the print counts in the original panda's technique, the overall event by time series has also been captured to text file for detailed investigation on this issue. Refer (Figure 24) on the overall report.

1	Datetime	Bufferpool	*DSNB319A (SHORTAGE)	*DSNB325A (CRITICAL)	DSNB327I (ADEQUATE)
2	2024-11-06 08:40:39	GBP5	0.0 1.0 0.0		
3	2024-11-06 08:40:39	GBP5-SEC	0.0 1.0 0.0		
4	2024-11-06 08:40:49	GBP5	0.0 0.0 1.0		
5	2024-11-06 08:40:49	GBP5-SEC	0.0 0.0 1.0		
6	2024-11-07 08:40:34	GBP5	0.0 1.0 0.0		
7	2024-11-07 08:40:34	GBP5-SEC	0.0 1.0 0.0		
8	2024-11-07 08:40:44	GBP5	0.0 0.0 1.0		
9	2024-11-07 08:40:44	GBP5-SEC	0.0 0.0 1.0		
10	2024-11-08 23:45:27	GBP4	1.0 0.0 0.0		
11	2024-11-08 23:45:27	GBP4-SEC	1.0 0.0 0.0		

Figure 24 GBP SHORTAGE overall report

## 9 Challaneges and discussions

Throughout the research, the author encountered various challenges and discussion points that merit further documentation to contribute to academic knowledge. These are outlined in the following subsections.

### 9.1 Steps incorporated to tackle challenges in source data analysis

Analyzing the syslog and identifying issues in the MSTR logs for Db2 on z/OS was indeed challenging, especially when dealing with unexpected or peculiar output in the MSTR JESMSG LG. Peculiar entries in the logs often signal deeper issues related to workload, configuration, or data problems. My Initial data analytics technique did not work as expected when I encountered few interesting challenges of display reports from Db2 system upgrade and operational activities.

Some of them are as follows:

- 1) DISPLAY BLOCKERS SUMMARY REPORT from -DIS BLOCKERS (DSNDB06, DSNDB01) DETAIL commands.
- 2) DISPLAY ARCHIVE REPORT with ARCHIVE LOG READERS reported in output.

I had to perform some manual cleanup for the blocker summary report and use data analytics technique to filter, refer (Figure 25). and delete 'BUSY CHC-LogR' lines which are from archive log readers.

```

import re

# Step 1: Read the log file into a list of strings
with open('/Workspace/Users/AB7028@student.jam.fi/zManav_Development/MSTRLOG.txt', 'r', encoding='iso-8859-1') as file:
    syslog_lines = file.readlines()

# Step 2: Exclude lines with specified text and write the remaining lines to a new file
with open('/Workspace/Users/AB7028@student.jam.fi/zManav_Development/MSTRLOG.txt', 'w', encoding='iso-8859-1') as outfile:
    for line in syslog_lines:
        if not re.search(r'BUSY CHC-LogR', line):
            # Write the line to the new file
            outfile.write(line)

```

Figure 25 Challenges with archive log readers

There are also other ways to avoid manual cleanup. New data analytics techniques can be used to identify the start of display report from DSNT360I (Figure 26) which has standard 35 asterisks before the report is printed and delete the lines until it find the end of report.

```

DSNT360I *****
          *****

```

### Explanation

This message consists of 35 asterisks that are part of the output from the DISPLAY DATABASE or DISPLAY BLOCKERS command. These commands are described in the messages [“DSNT361I” on page 561](#) and [“DSNT369I” on page 570](#).

Figure 26 DSNT360I with 35 asterisks DISPLAY REPORT

## 9.2 Limitations with in Databricks

The author encountered some challenges within databricks while processing huge Db2 MSTR SYSLOG text as well as TSV file. Two main challenges were, the author was unable to view the processed very large TSV file with in databricks, refer (Figure 27 ) and also unable to export the TSV file, refer (Figure 28 ).Author’s believe, once zManav starts utilizing aws s3 on full scale operation, ,

we can overcome these limitations.

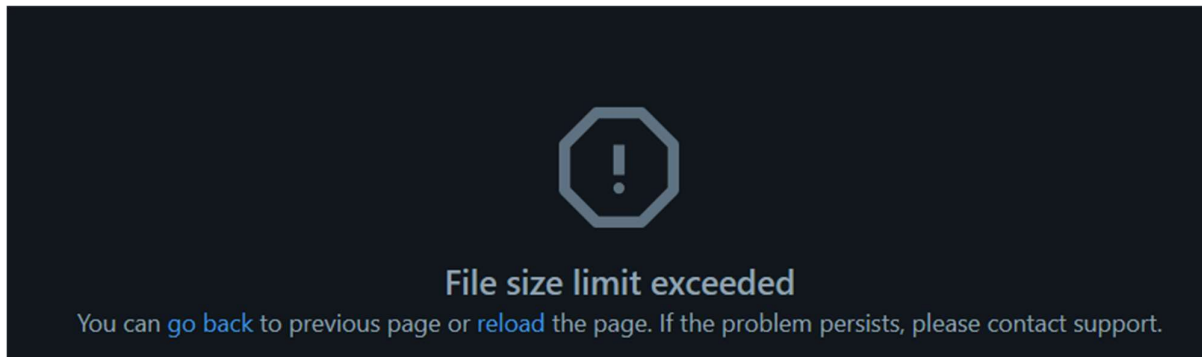


Figure 27 TSV file size limit exceeded.

```

1  {
2    "error_code": "BAD_REQUEST",
3    "message": "content size (37069530) exceeded the limit 10485760"
4  }

```

Figure 28 Bad request unable to export TSV

### 9.3 Further research expansions

In all above cases, the key is Db2 MESSAGE ID, IBM Db2 for z/os introduces a set of system messages (IBM 2023). to assist database administrators and users in diagnosing and resolving issues. Each message is paired with a unique message ID, which is a critical tool for identifying specific errors or events. These IDs provide a reference to documentation and support resources, enabling users to quickly address problems. The message text typically includes a description of the issue, along with potential causes and solutions, making it essential for effective troubleshooting and system management.

Author's aim to integrate Db2 V13 message pdf in the data analysis part to make the tool as one stop shop for administrators for proactive and reactive root cause analysis of any situation.

For situations where there is a new message ID, the tool is expected to extract the explanation from reference manuals that are used as plugin for quick references. IBM URL scrapping is unfortunately unavailable.

Refer (Figure 12) as an example how the MESSAGE ID description are documented in PDF given as reference above.

<b>DSN7507I</b>	<b><i>csect-name</i> SCA STRUCTURE <i>sca-structure-name</i> IS ALLOCATED IN A VOLATILE STRUCTURE.</b>
<b>Explanation</b>	The shared communications area (SCA) structure is volatile.
<b>System action</b>	Db2 processing continues.
<b>System programmer response</b>	One Db2 issues this message whenever the structure has the volatile attribute. Db2 detects this either on the first connection to the SCA or on the attribute changes later. Db2 always requests a nonvolatile SCA. If this is not what you want, change the MVS CFRM administrative policy and request a rebuild of the SCA with the MVS SETXCF START,REBUILD command.

Figure 12 Db2 MESSAGE ID example

## 9.4 Mainframe Root Cause Analysis

In IT world, the incidents are typically categorized based on their impact and urgency:

Severity Levels:

- P1 (Critical): Major system outages, causing significant disruption.
- P2 (High): Serious issues that impact performance but do not stop operations.
- P3 (Medium): Minor disruptions or degraded performance.
- P4 (Low): Small issues with minimal impact on business operations.

Mainframe infrastructure root cause analysis (RCA) scenarios typically involve investigating system failures, performance issues, or outages to determine the underlying cause. Common scenarios include hardware failures, software bugs, network issues, or misconfigurations. RCA aims to identify the source of problems like CPU overutilization, memory leaks, or disk space issues, which can disrupt critical operations. By systematically analyzing logs, performance metrics, and system behaviours, teams can implement corrective actions, ensuring system stability and preventing future occurrences. Effective RCA requires knowledge of system components, diagnostic tools, and troubleshooting methodologies.

In Major Incident Management (MIM), the situation is often intense when a critical incident occurs. The MIM team, including IT operations, support staff, and key stakeholders, is quickly mobilized to address the issue. The goal is to minimize downtime and mitigate business impact while identifying the root cause of the incident. Teams communicate continuously, balancing urgency with methodical troubleshooting. Tension is high, as the clock ticks to resolve the incident, often requiring cross-departmental collaboration and effective decision-making to restore services and prevent further escalation.

zManav will play a pivotal role in Root Cause Analysis (RCA) by providing deep insights into system patterns, user interactions, and potential anomalies. It will help quickly identify the source of issues, reduce troubleshooting time, and enhance the overall effectiveness of the RCA process. This tool ensures a more accurate and efficient analysis, minimizing downtime and enabling faster resolutions.

## **10 Conclusions**

This research examined the impact of advanced analytics-based syslog analysis on mainframe performance monitoring, specifically focusing on how techniques can improve system reliability and reduce incident response times. The findings indicate that data analytics significantly enhance the ability to detect performance anomalies quickly, enabling faster incident resolution. This reduction in downtime is crucial for organizations that rely on mainframe systems for their most critical operations. Moreover, the automation of log analysis contributes to increased efficiency by eliminating the need for manual intervention in routine monitoring tasks.

In conclusion, analytics-based syslog analysis offers identifying significant opportunities for improvements in both reliability and operational efficiency for mainframe Db2 systems, providing businesses with a proactive and efficient solution to meet their performance monitoring needs.

By addressing the research questions (RQ1,RQ2,RQ3) with precise outputs, I am optimistic that this tool will provide a groundbreaking solution for Mainframe Administrators in root cause analysis investigations.

Given its cost-effectiveness and the growing AI demand, I firmly believe that zManav will bring a significant revolution to the mainframe monitoring platform. This research is expected to undergo substantial expansion, driven by increasing requirements and remarkable enhancements such as machine learning/neural network, particularly with the integration of AI plugins such as chatbots. While there is still room for improvement, this research has been a dream of mine for many years—a vision I've held close to my heart. I envisioned this mechanism being in place during challenging times, especially when investigating problems during major crises.

zManav has successfully fulfilled its core objective by enabling trend analysis and generated alerts on new error messages that have never been encountered in mainframe Db2 for z/OS.

With 18+ years technical experience as Mainframe Db2/IMS systems engineer and as an author of this thesis, I estimate the tasks that previously required extensive manual sorting, filtering, and computation are now automated. Hence practically, this research work delivers a tool that improves operational efficiency by reducing manual syslog processing time by 50% to 60% and achieving high anomaly detection accuracy, thereby offering significant value to system administrators in managing complex mainframe environments effectively.

Future enhancements will include the data analytics technique deployed on full-scale using aws s3, Athena and power BI. In addition, author will use IBM PDF's/URLs for quick reference to Message ID explanations which will further broaden the tool's capabilities.

## References

Apollo11Space. (n.d.). *IBM's System/360: The groundbreaking mainframe that helped land men on the Moon*. Retrieved November 18, 2024, from <https://apollo11space.com/ibms-system-360-the-groundbreaking-mainframe-that-helped-land-men-on-the-moon/>

Astecno. (2023). *ITIC 2023 Global Server Hardware and Server OS Reliability Report*. Retrieved from <https://astecno.com.br/wp-content/uploads/2023/09/ITIC-2023-Global-Server-Hardware-Server-OS-Reliability-Report.pdf>

Business Research Insights. (2024, November 11). *Mainframe sales market size, share, and trends report 2024–2032*. Retrieved November 28, 2024, from <https://www.businessresearchinsights.com/market-reports/mainframe-sales-market-112563>

Dave's Garage. (2023, October 8). *Why Do Mainframes Still Exist? What's Inside One? 40TB, 200+ Cores, AI, and more!* [Video]. YouTube. <https://www.youtube.com/watch?v=ouAG4vXFORc>

Deloitte. (n.d.-a). *Mainframe talent: Addressing the looming talent shortage* [Infographic]. Deloitte. Retrieved November 20, 2024, from <https://www2.deloitte.com/content/dam/Deloitte/us/Documents/about-deloitte/us-mainframe-talent-infographic.pdf>

Deloitte. (n.d.-b). *What we do*. Deloitte. Retrieved November 20, 2024, from [https://www.deloitte.com/global/en/what-we-do.html?icid=top\\_what-we-do](https://www.deloitte.com/global/en/what-we-do.html?icid=top_what-we-do)

European Banking Authority (EBA). (n.d.). *Digital Operational Resilience Act (DORA)*. European Banking Authority. Retrieved from <https://www.eba.europa.eu/activities/direct-supervision-and-oversight/digital-operational-resilience-act>

IBM. (2023). *Db2 13 for z/OS: Messages (SSEPEK\_13.0.0)*. IBM. [https://www.ibm.com/docs/en/SSEPEK\\_13.0.0/pdf/db2z\\_13\\_msgsbook.pdf](https://www.ibm.com/docs/en/SSEPEK_13.0.0/pdf/db2z_13_msgsbook.pdf)

IBM. (2023, October 3). *Db2 address spaces in the z/OS environment*. Retrieved from <https://www.ibm.com/docs/en/db2-for-zos/13?topic=architecture-db2-address-spaces-in-zos-environment>

IBM. (n.d.-a). *Database management systems in the z/OS environment*. Retrieved November 18, 2024, from <https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-database-management-systems-in-environment>

IBM. (n.d.-b). *Adopting new capabilities in Db2 13 continuous delivery*. Retrieved from <https://www.ibm.com/docs/en/db2-for-zos/13?topic=adopting-new-capabilities-in-db2-13-continuous-delivery>

IBM. (n.d.-c). *Who uses mainframes and why do they do it?* Retrieved November 18, 2024, from <https://www.ibm.com/docs/en/zos-basic-skills?topic=vmt-who-uses-mainframes-why-do-they-do-it>

IBM. (n.d.-d). *Db2 address spaces in the z/OS environment*. Retrieved November 18, 2024, from <https://www.ibm.com/docs/en/db2-for-zos/12?topic=architecture-db2-address-spaces-in-zos-environment>

IBM. (n.d.-e). *Who is a system programmer?* Retrieved November 18, 2024, from <https://www.ibm.com/docs/en/zos-basic-skills?topic=world-who-is-system-programmer>

IBM. (n.d.-f). *Roles in the mainframe world today*. Retrieved November 18, 2024, from <https://www.ibm.com/docs/en/zos-basic-skills?topic=today-roles-in-mainframe-world>

IBM. (n.d.-g). *What is JES?* Retrieved November 18, 2024, from <https://www.ibm.com/docs/en/zos-basic-skills?topic=jobs-what-is-jes>

IBM. (n.d.-h). *Coupling facility structure names*. In *Db2 13 for z/OS documentation*. Retrieved November 25, 2024, from <https://www.ibm.com/docs/en/db2-for-zos/13?topic=sharing-coupling-facility-structure-names>

IBM. (n.d.-i). *The role and importance of Db2 group buffer pools in data sharing groups*. IBM Documentation. Retrieved November 28, 2024, from <https://www.ibm.com/docs/en/om-db2-pe/5.4.0?topic=overview-role-importance-db2-group-buffer-pools-in-data-sharing-groups>

IBM Developer. (n.d.). *What is IBM Z?*. Retrieved November 18, 2024, from <https://developer.ibm.com/articles/what-is-ibm-z/>

IBM Documentation. (n.d.-a). *z/OS: What is REXX?* IBM. Retrieved November 16, 2024, from <https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-rexx>

IBM Documentation. (n.d.-b). *Db2 for z/OS 13 - Messages*. IBM. Retrieved November 16, 2024, from <https://www.ibm.com/docs/en/db2-for-zos/13?topic=db2-messages>

IBM Documentation. (n.d.-c). *\*DSNJ002I\**. IBM. Retrieved November 16, 2024, from <https://www.ibm.com/docs/en/db2-for-zos/12?topic=messages-dsnj002i>

IBM Documentation. (n.d.-d). *\*Active and archive logs\**. IBM. Retrieved November 16, 2024, from <https://www.ibm.com/docs/en/zos-basic-skills?topic=objects-active-archive-logs>

IBM zSystem. (n.d.). IBM - United States. <https://www.ibm.com/history/eserver-zseries>

Kela. (2022, May 30). *Kela hankkii keskuskoneteknologiati IBM:ltä – hankinnan arvo noin 60 miljoonaa euroa*. Kela. Retrieved November 21, 2024, from <https://www.kela.fi/ajankohtaista/kela-hankkii-keskuskoneteknologiati-ibm-lta-hankinnan-arvo-noin-60-miljoonaa-euroa>

National Association of State Chief Information Officers (NASCIO), & Accenture. (2021). *Cloud study: Accelerating digital transformation in the public sector*. Retrieved from <https://www.nascio.org/wp-content/uploads/2021/10/NASCIO-Accenture-Cloud-Study.pdf>

Peppers, K., Tuunanen, T., Gengler, C. E., Rossi, M., Hui, W., Virtanen, V. & Bragge, J. (2006). The Design Science Research Process : A Model for Producing and Presenting Information Systems Research. In *1st International Conference, DESRIST 2006 Proceedings*. (pp. 83-106). Claremont Graduate University. <http://urn.fi/URN:NBN:fi:jyu-201904092111>

Sanatana Dhara. (n.d.). *Manav – Human being*. Retrieved November 18, 2024, from <https://sanatanadhara.com/manav-human-being/>

TechChannel. (n.d.). *IBM z16 and Power10 deliver highest reliability among mainstream servers for 15th consecutive year*. Retrieved from <https://techchannel.com/backup-and-recovery/ibm-z16-and-power10-deliver-highest-reliability-among-mainstream-servers-for-15th-consecutive-year/>

ZMainframes. (2024, November 18). *Mainframe computer and how is it different from other computers?* ZMainframes. Retrieved November 18, 2024, from <https://zmainframes.com/zlog/mainframe-computer-and-how-is-it-different-from-other-computers/>

Sbai Idrissi, Y. (2023, July 18). *Overcoming the toughest challenges in IBM Db2 migration*. IBM Community. Retrieved November 18, 2024, from <https://community.ibm.com/community/user/datamanagement/blogs/youssef-sbai-idrissi1/2023/07/18/overcoming-the-toughest-challenges-in-ibm-db2-migr>

Northeastern University. (2020, August 7). *Data Analysis Project Lifecycle*. Retrieved from <https://graduate.northeastern.edu/resources/data-analysis-project-lifecycle/>

Precisely. (2022). *9 mainframe statistics that may surprise you*. Retrieved from <https://www.precisely.com/blog/mainframe/9-mainframe-statistics>

Yle Uutiset (2024,3). *Tässäkö syy pankkiongelmiiin? Tietojärjestelmät ovat 80-luvulla käyttöönotetun keskustietokoneen varassa*((2024, October 16). Yle Uutiset. <https://yle.fi/a/74-20118006>

ZMainframes. (n.d.). *The role of mainframe in India's Chandrayaan-3 mission*. Retrieved November 18, 2024, from <https://zmainframes.com/zlog/the-role-of-mainframe-in-indias-chandrayaan-3-mission/>

## Appendices

### Appendix 1. Data Management Plan

Data management plan is available in <https://dmptuuli.fi/plans/26321>

#### Plan Overview

A Data Management Plan created using DMPTuuli

Title: zMANAV

Creator: Manigandan Ravichandran

Principal Investigator: Rantonen Mika, Sipola Tuomo

Affiliation: Jamk University of Applied Sciences

Template: Data management plan for theses at Jamk University of Applied Sciences

Project abstract:

With over 18 years of experience as a mainframe infrastructure administrator, I have gained an in-depth understanding of the complexities and critical nature of mainframe operations. This experience has inspired the main goal of my thesis: the development of a tool named "zManav." The lowercase 'z' in "zManav" is deliberately chosen to align with IBM's naming conventions, where 'z' signifies 'zero downtime,' reflecting the tool's aim to enhance system reliability and availability. "Manav," derived from the Sanskrit word for "student" or "human," represents the tool's design philosophy. It embodies the concept of a learning, intelligent entity within the artificial intelligence paradigm,

akin to a student or human learning and adapting in the realm of mainframe technology. This tool is intended to simplify the challenges faced by mainframe administrators in an era where data is as valuable as oil. IBM's Db2 for z/OS, alongside IBM IMS, offers a highly reliable, secure, and scalable database environment on the mainframe platform, and is the most used database in this context. The centerpiece of my Master's degree thesis, zManav, aims to revolutionize how Mainframe subsystems like DB2, MQ, IMS, zOS systems behaviour can be monitored and analyzed from the operational logs. It focuses on the software aspect of the DB2 subsystem, excluding customer or application data, to address issues such as deadlocks, timeouts, potential message ID with warnings and resource unavailability and so on. Through periodic automated reporting facilitated by a graphic tool, zManav aspires to streamline and improve the efficiency of mainframe administration. The thesis also incorporates machine learning from previously generated message and message ID to recognize patterns and identify critical system messages. Additionally, it provides alerts when new message arrive that may be system-critical and require potential action from system programmer. Moreover, the author, recognizing the broader potential of this tool, has plans to expand its capabilities beyond DB2 systems.

For example:

- MQ (Message Queue) for z/OS.
- IMS (Information Management System database) for z/OS.
- CICS (Customer Information Control System) for z/OS.
- z/OS (z Mainframe Operating System syslogs)

ID: 26321

Start date: 01-01-2024

End date: 31-05-2025

Last modified: 02-12-2024

Created using DMPTuuli. Last modified 02 December 2024 1 of 3

## zMANAV

### 1. General description of data

Describe, what kinds of data is your research based on? What data will be collected, produced, or reused? What file formats will the data be in?

Mainframe Db2 functions as a formal subsystem within the z/OS environment, efficiently interacting with other z/OS subsystems and components. It operates as a distinct relational Database Management System (DBMS) instance, managing the creation, organization, modification, and access to database data. Db2 for z/OS processes execute in several z/OS address spaces, including a) the Database Services Address Space (ssnmDBM1), responsible for most database-related services; b) the System Services Address Space (ssnmMSTR), overseeing the starting, stopping, and local access control of Db2; and c) the Distributed Data Facility (DDF) Services Address Space (ssnmDIST), supporting network communications and remote database access requests. Additionally, Db2 uses d) the Internal Resource Lock Manager (IRLM) Address Space (irlmproc) for database resource control and locking, alongside other specialized address spaces for various functions. In this research, I will meticulously collect DB2 Master Task (MSTR) logs on a daily basis to perform an in-depth analysis of the DB2 system's health and behaviour. This process involves extracting relevant data from these logs and save in AWS S3 bucket to load them into Databricks as TXT file (or) directly upload in Darabricks and convert to TSV (Tab Separated Value) file using Pandas Data frame Data Analytics . Considering the fact that Mainframe subsystems log/ syslogs are typically in a format that is not easily readable by data analytics tools. By converting them to TSV (tab-separated values) files, we can transform the data into a structured format that is easier to analyze. TSV files can be easily imported into spreadsheets or data analysis tools , allowing us to perform more advanced analysis on the data. Additionally, TSV files can be compressed to reduce storage space, making it easier to store and manage large amounts of data. The converted TSV files are further used by for variety of data analytics techniques to gain more insights and make data-driven decisions such as a) Trend analysis; b)Anomaly detection;c)Predictive modeling by using machine learning algorithms, we can build models that can predict future events or behaviours based on historical data.It is also intended to maintain the historical data by storing and reusing processed TSV file to predict overall system behaviour trends on demand. How will the consistency and quality of data be controlled?

Mainframe Syslogs, as well as DB2 and MQ started tasks , keep track of all logs from the time they are started until they are shut down in Mainframe server. The Consistency and quality of the data will be controlled data timestamps while processed and backedup in databricks or aws S3 where I perform it in the form of TSV file for further data analytics techniques.

### 2. Ethical and Legal Compliance

What ethical and legal issues are related to your data management, for example, the Data Protection Act and other legislation related to the processing of the data? Do you process personal data

(yes/no)?

Appropriate approvals are secured from OP bank database product owner to undertake the thesis using the Mainframe software logs . In this particular case, I will be focusing on the Mainframe DB2 logs and the techniques will be further developed for other subsystems including z/OS syslog's. Due to organization's Data Protection regulations, the input logs or processed outputlogs shall not be saved in any personal media , public domain or public cloud other than OP specific intranet storage. The thesis is performed using TEST (Non-prod) software logs and when matured, the author shall deploy it considering the techniques for production implementation that is limited for OP bank use only .

No personal data is available in Mainframe software logs and not applicable in this reserach

Data access rights? Is the data confidential?

Data is accessible only authorized personal in OP. The software logs used for research were processed within OP network and can not be shared in public domain . Only the metrics of message ID from research outcome is key and it is well described and explained for thesis.

### 3. Documentation and metadata

How do you document and describe your data

My documentation will be in the jamk thesis template in english and appropriate Mainframe related technical background will be described for user's to get an overview and moto on the research involved.

### 4. Storage and backup during the thesis project

Where will your data be stored, and how will it be backed up?

All syslogs that are in scope of the research will be stored in OP intranet/devices such as op databricks,aws s3 bucket and my OP laptop shall be used as a medium for transfers between the source and target repositories. With every IPL (Mainframe restart) appropriate backups are planned part of my reserach technique in data analysis and they shall be maintained with in databricks or aws s3 bucket.

Who will be responsible for controlling access to your data, and how will secured access be controlled?

My organization, OP bank has dedicated security team who control access using our AD account and I have full authority to manage my own data in the repositories.

Created using DMPTuuli. Last modified 02 December 2024 2 of 3

### 5. Archiving and opening, destroying or storing the data after the thesis project

Can some of the data be made openly available and published? Where will the data be published?

The input data in TXT format and processed output data in TSV format shall not be available for public . However detailed description on the techniques, data analytics derived using Panda dataframe and staistical output figures shall be made available in thesis document as evidences for further publishing in theseus as final report.

Where will data be stored, and for how long? How is the data destroyed?

The data used in research shall be stored with no expiry and in an event if unnessary data need to be destroyed , I shall delete them from my organization respositories which is safe and trusted source .

### 6. Data management responsibilities and resources

Who will be responsible for specific tasks of data management during the life cycle of the research project? Estimate the resources.

I will be the only person responsible for the end to end data management during the lifecycle of my research project (or) for any further developments.

Created using DMPTuuli. Last modified 02 December 2024 3 of 3