

Importance of Mainframe in Modern Era of Technologies

Subhani Shaik

Programmer Analyst, Conch Technologies Inc, TN, USA

ABSTRACT

Legacy mainframe programming languages have been foundational in the development and operation of large-scale business and scientific computing systems since the mid-20th century. These languages, including COBOL, Assembler, PL/I, FORTRAN, RPG, JCL, ALGOL, Natural, CLIST, and REXX, were designed to handle the robust data processing needs of enterprises and government institutions. COBOL, known for its English-like syntax, is still widely used in business applications, while Assembler offers low-level hardware control for highly efficient execution. PL/I and FORTRAN cater to both business data processing and scientific computations, respectively. RPG, originally created for report generation, evolved to support comprehensive data processing tasks. JCL, although a scripting language rather than a programming language, is crucial for job execution in batch processing environments on IBM mainframes. ALGOL has been influential in the development of many modern languages. Natural and REXX facilitate rapid application development and scripting, respectively. Despite the emergence of modern programming languages, these legacy languages remain vital for maintaining and understanding existing mainframe systems, underscoring their enduring significance in the history of computing.

*Corresponding author

Subhani Shaik, Programmer Analyst, Conch Technologies Inc, TN, USA.

Received: January 06, 2024; **Accepted:** January 14, 2024; **Published:** January 26, 2024

Keywords in Legacy Mainframe

Here are some key words commonly associated with mainframe programming languages.

Keywords: COBOL (Common Business-Oriented Language), Assembler, Assembly Language, PL/I (Programming Language One), FORTRAN (Formula Translation), RPG (Report Program Generator), JCL (Job Control Language), ALGOL (Algorithmic Language), Natural, CLIST (Command List), REXX (Restructured Extended Executor), MVS (Multiple Virtual Storage), TSO/E (Time Sharing Option/Extensions), CICS (Customer Information Control System), IMS (Information Management System), VSAM (Virtual Storage Access Method), DB2, Adabas, Batch Processing, Data Processing, Mainframe, Legacy Systems, High-Level Language, Low-Level Language, Procedural Programming, Structured Programming, Batch Jobs, Scripting, Automation, Enterprise Computing, Regulatory Compliance, System Programming, Data Management, Job Scheduling, Resource Management

These keywords encompass the critical features, functionalities, and benefits of aviation maintenance software, reflecting its importance in managing and optimizing aircraft maintenance operations.

Introduction to Legacy Mainframe Programming Languages

Legacy mainframe programming languages have been instrumental in shaping the landscape of enterprise computing since the mid-20th century. These languages were developed to meet the robust data processing and operational needs of large organizations, including businesses, government agencies, and scientific institutions. Mainframes, known for their reliability, scalability, and capacity to handle vast amounts of data, have relied on these

programming languages to perform complex computations, manage large-scale transactions, and ensure operational continuity.

Historical Context and Evolution

Mainframe computers emerged in the 1950s and 1960s, providing a significant leap in computational power and storage capacity compared to earlier systems. The need for efficient and reliable software to harness this power led to the development of several specialized programming languages. These languages were designed to optimize the performance of mainframe systems, manage extensive data sets, and automate a wide range of business processes.

Key Legacy Mainframe Programming Languages

COBOL (Common Business-Oriented Language)

- **Overview:** Developed in the late 1950s, COBOL is designed for business data processing. Its English-like syntax makes it readable and maintainable, facilitating widespread adoption in financial institutions and government systems.
- **Applications:** Payroll processing, transaction processing, and administrative systems.

Assembler (Assembly Language)

- **Overview:** Assembly language provides low-level control over hardware, enabling highly efficient execution of programs. It requires detailed knowledge of the computer's architecture.
- **Applications:** System programming, performance-critical applications, and hardware interfacing.

PL/I (Programming Language One)

- **Overview:** Introduced in the 1960s, PL/I combines features of

both business-oriented and scientific programming languages, making it versatile for a variety of applications.

- **Applications:** Business data processing, scientific computations, and system programming.

FORTRAN (Formula Translation)

- **Overview:** One of the oldest high-level languages, FORTRAN was developed for scientific and engineering calculations. It excels in numerical computation and array operations.
- **Applications:** Scientific research, engineering simulations, and high-performance computing.

RPG (Report Program Generator)

- **Overview:** Originally created for generating business reports, RPG has evolved to handle comprehensive data processing tasks. It is particularly popular on IBM midrange and mainframe systems.
- **Applications:** Business reporting, data analysis, and enterprise resource planning (ERP) systems.

JCL (Job Control Language)

- **Overview:** JCL is a scripting language used to instruct mainframes on how to execute batch jobs. It is essential for managing job execution, resource allocation, and output handling.
- **Applications:** Batch processing, job scheduling, and system operations.

ALGOL (Algorithmic Language)

- **Overview:** ALGOL, developed in the 1950s, influenced many modern programming languages. It introduced concepts like structured programming and recursion.
- **Applications:** Academic research, algorithm development, and early system programming.

Natural

- **Overview:** A fourth-generation language (4GL) developed by Software AG, Natural is integrated with the Adabas database management system, facilitating rapid application development.
- **Applications:** Business applications, database management, and application development.

CLIST (Command List)

- **Overview:** CLIST is a procedural scripting language used on IBM mainframes for automating tasks in the TSO/E environment.
- **Applications:** Scripting, task automation, and system administration.

REXX (Restructured Extended Executor)

- **Overview:** REXX is an interpreted scripting language developed by IBM. It is known for its ease of use and versatility in both procedural and structured programming.
- **Applications:** Scripting, automation, and prototyping.
- **Importance and Legacy**

These legacy mainframe programming languages have played a critical role in the development of reliable and efficient systems that have supported various industries for decades. Despite the advent of modern programming languages and technologies, many of these legacy languages remain in use today, particularly in maintaining and updating existing mainframe systems. Their continued relevance underscores their robustness, efficiency, and the significant investment organizations have made in these

technologies over the years.

Understanding these languages is essential for maintaining legacy systems, transitioning to new technologies, and appreciating the historical development of computing. They represent the foundation upon which much of today’s enterprise and large-scale computing infrastructure is built.

Table 1 Provides a summary of the similarities and differences between mainframe and distributed platforms.

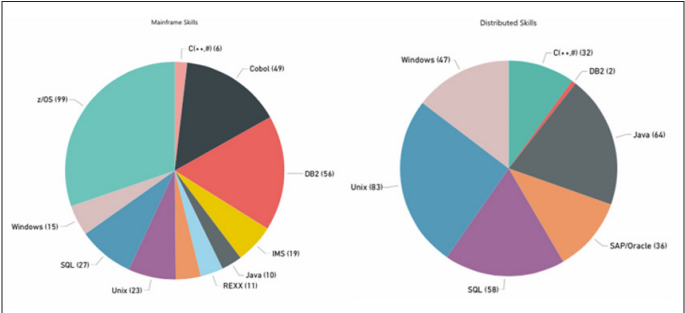
Table 1: Mainframe vs Distributed

	Mainframe	Distributed
Operating System	z/OS, Linux	Linux, Windows
Processors	Up to 141 per server*	Up to 4†
Memory	Up to 10 TB*	Up to 6TB†
Storage	External	External/Internal
Transactions per Minute	1,700,000*	6,602†
Scaling Method	Vertical (within system)	Horizontal (more servers)
Programming Languages	C++ Java Cobol Rexx Assembler PL/I CLIST	C++ Java - - - - Perl C# Python
Database Systems	DB2 IMS Hadoop	- - Hadoop SQL

*Based on z13. Transactions based on theoretical 2.5 billion transactions per day.
†Based on HP Proliant DL580 Gen9 with 4 Intel Xenon E7-8890 v3 processors

The mainframe is, however, facing a serious issue. The average age of mainframe workers is 55 to 60 and most will be entering retirement soon. The issue is a result of the fact that computer science curriculums at most colleges do not cover many mainframe technologies or programming languages because the platform is viewed as being old technology. Students of computer science have little desire to learn the old programming languages that run the mainframe, instead preferring to focus on newer technologies. This trend is putting pressure on mainframe departments at businesses that rely on the technology to run critical business applications. This is evident in the context of the findings from the Platform Skill Comparison Analysis below.

Below figure represent the breakdowns of the desired skills for the mainframe and distributed platforms, respectively. Keep in mind that each number next to the skill represents how many jobs how of the 101 reviewed for each platform listed that skill as necessary for the job.



The results were close to what was expected. 99% of the jobs for the mainframe asked for experience with z/OS while the remaining two jobs asked for familiarity with a different mainframe-only OS. On distributed systems, Unix was the most asked for operating system, with 83% of jobs wanting experience with mainly Linux. Moving to programming languages, 64% of jobs on distributed platforms asked for Java experience compared to only 10% of mainframe jobs. COBOL also seemed important to know for a

mainframe position, with 49% of jobs listing that as a desired skill, but on distributed platforms it was not mentioned. 6% of mainframe jobs wanted experience with C languages, while 32% of distributed jobs wanted experience with the language.

When it comes to which database systems employers are looking for experience with, there is more overlap between the two platforms, but it is still apparent which ones are favored on each. Experience with SQL was by far the most common database skill sought on both platforms with 27% of mainframe and 54% of distributed jobs requiring skills with SQL. DB2 was by far the most common database system that mainframe jobs were seeking, with 56% of employers asking for experience. Just two distributed platform positions wanted experience with DB2.

While there is overlap between the platforms for some of the required skills, there are many that are platform specific, especially programming languages and operating systems. This means that potential hires for these jobs need to have platform specific knowledge. Students at universities need to decide which platform they want to work with in order to acquire these skills or businesses will need to provide on the job training for specific skills.

For students that wish to pursue an education in mainframe technology, they will find it harder to find a school that offers courses on the mainframe compared to distributed platforms. IBM has begun to combat this skills gap, along with several mainframe software companies, such as CA Technologies. IBM began their System Z Academic Initiative in 2003 at 24 colleges, which in 2010 had expanded to 700 colleges. This program introduces students to mainframe technology and allows them to get hands-on experience by providing access to mainframes. Companies such as Citi and Bank of America also visit campuses to talk to students about job opportunities in the mainframe field.

Mainframe software companies are trying to solve the issue of retiring mainframe workers by designing new ways to interact with the mainframe. CA Technologies, for example, is working hard to make their products more appealing to the next generation. Millennials are much more comfortable with graphical interfaces because that is what they have used for their entire lives. CA is developing newer versions of their programs that utilize a GUI instead of relying on a classic mainframe green screen.

While the mainframe may be thought of as a legacy platform, in the new era of big data, the mainframe is gaining new roles. Data analytics is becoming an important way to plan business activities. How do companies process the large amounts of data that they collect to find trends? The mainframe is a very likely candidate. For example, WalMart has been using mainframes to track real-time inventory data as sales happen in each of its stores to plan shipments to stores. This high-tech supply chain has allowed Wal-Mart to meet customer demand by having the right products in the right stores at the right time. Visa and Citi have been able to take advantage of the mainframe's ability to perform data analytics in real time to detect fraud as a transaction is processed. This helps reduce their losses due to fraud because a transaction can be rejected right away instead of performing fraud analysis after the fact.

Conclusion

The mainframe may not be the right fit for every company but remains a viable modernized platform for many large businesses today. Small businesses that do not have to run large datacenters probably would not find a benefit by using a mainframe. For large

IT shops, though, the mainframe still serves an important purpose. It has gained new abilities that make it attractive for first time buyers and for the many companies that have used mainframes in the past, they continue to embrace the mainframe as a large part of their IT operations. The mainframe's ability to provide the transactional backbone for banks and insurance companies along with the new abilities it has gained in recent years allow the mainframe to remain relevant in modern business [1-4].

References

1. (2014) Walmart Delivers Customer Service with the IBM Mainframe. IBM.
2. Willmott Don (2012) Take Advantage of the Mainframe Talent Shortage - Dice Insights. Dice <https://www.dice.com/career-advice/mainframe-talent-shortage>.
3. Rachael K (2010) Big Tech Problem as Mainframes Outlast Workforce. Bloomberg <https://www.bloomberg.com/news/articles/2010-08-03/big-tech-problem-as-mainframes-outlast-workforcebusinessweek-business-news-stock-market-and-financial-advice>.
4. Clabby Joe (2012) The IT Skills Shortage: Where Will We Find Mainframe Talent? SHARE: Blogs <https://blog.share.org/Technology-Article/it-skills-shortage-where-will-we-find-mainframe-talent>.

Copyright: ©2024 Subhani Shaik. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.