MAINFRAME MORE RELEVANT THAN EVER TO CORE FINSERV APPS

The modern mainframe is foundational to strategic banking and finance projects, including API-first banking as a service, open banking, generative artificial intelligence (GenAI) model training, and embedded finance opportunities.

Mainframe applications operate core mission-critical applications for leading financial services firms, such as core banking, claims processing, and payment processing. Such applications support the global economy.

Recent advances in mainframe hardware and software bring the platform into the 21st century, making it easier than ever to include mainframes in strategic IT projects and simplify their operation.

Mainframe managed data is also a critical source for a growing number of banking and finance GenAI applications, such as customer service, sales and marketing, fraud detection, cybersecurity, and regulatory reporting. Mainframe data from mission critical applications is a veritable goldmine for LLM training purposes.

The modern mainframe also helps address organizational challenges such as technical debt and a retiring mainframe workforce, which enable mainframe applications to participate equally in .

trategic financial services initiatives leverage all of these infrastructures, and new hardware and software technologies for the mainframe enable such initiatives.

Mainframe evolution

The mainframe computer has a long and successful history in the IT industry. The first commercial applications on mainframes had a tremendous impact on the financial services industry, successfully automating previously manual operations..

Advances in technology enabled new types of applications for distributed computers, PCs, and handheld devices, but core mission-critical applications on the mainframe still anchored IT infrastructure. the mainframe. Mainframes remain the best platform for high volume, reliable transaction processing.

Modern mainframes offer advanced CPUs, virtual machines, modern operating systems (i.e. Linux), modern programming languages (e.g. Java and Python), and modern software tools (e.g. CI/CD pipelines and AIOps), and provide many capabilities initially created for distributed and cloud computing environments (e.g. Docker containers). All of these new capabilities testify to the mainframe's continued success and help support new strategic initiatives.

The 2025 BMC Mainframe Survey shows that a majority of financial services organizations continue to run mission-critical applications on mainframes and continue to invest in those applications with the goal of modernizing and supporting new initiatives.

Modernizing mission critical mainframe applications also helps reduce overall cost, can improve uptime and resiliency characteristics, and helps mitigate the larger risk of migration.

Significant mainframe characteristics

A computer that originally took up an entire room is now about the size of a refrigerator. That reduced size packs significantly more power than the original models, and at a lower cost.

Over the years, mainframes just kept getting faster and faster and costing less and less.

It makes more sense than ever to invest in applications for the mainframe, especially high-volume, reliable transaction processing applications such as core banking and claims processing. The mainframe architecture is especially well-suited for this type of application.

And the resulting transactional data records can be incredibly important for feeding decision support, analytic, and GenAI systems.

Processing transactions, typically by coordinating two or more operations on data into a single unit that succeeds or fails as a unit (such as a debit/credit operation performed for a simple funds transfer, or a more complex ATM withdrawal operation), originated on the mainframe and arguably still works best on the mainframe.

```
Begin Transaction Transfer

Write Debit Amount to Account A

Write Credit Amount to Account B
```

```
<error>

End Transaction
```

Transactional consistency—meaning operations on data within a transaction (as in the code snippet above) succeed or fail as a unit—is impacted by the latency of distributed computing.

The time and distance to and from the disk taken to coordinate multiple operations is impacted by network latency, especially when a large workload is divided across multiple computers in a cluster or cloud. When the entire application is on the mainframe, that latency is reduced and the performance and reliability is increased.

This kind of transactional consistency and accurate recording of results is especially important for core financial services applications. At the end of the day everything must balance to the proverbial penny.

Mainframes have also evolved to become first class participants in hybrid cloud architectures that are increasingly common in the financial services world for analytics, reporting, and GenAI, due to their ability to handle and transfer large volumes of data reliably.

Standardized interfaces

While I was Chief Architect for the Investment Banking Division of a multinational investment bank, a strategic program we ran was called Managed Evolution.

The idea was to use service-oriented architecture (SOA) interfaces to encapsulate existing functionality, connect it to new applications, and provide consistent access to the applications from GUIs. Originally, we used CORBA but we evolved the implementation to include SOAP and REST.

The main objective is to have consistent, standard APIs and data models available enterprise-wide. (The implementation format and communication protocol can vary.) This illustration visualizes the idea:

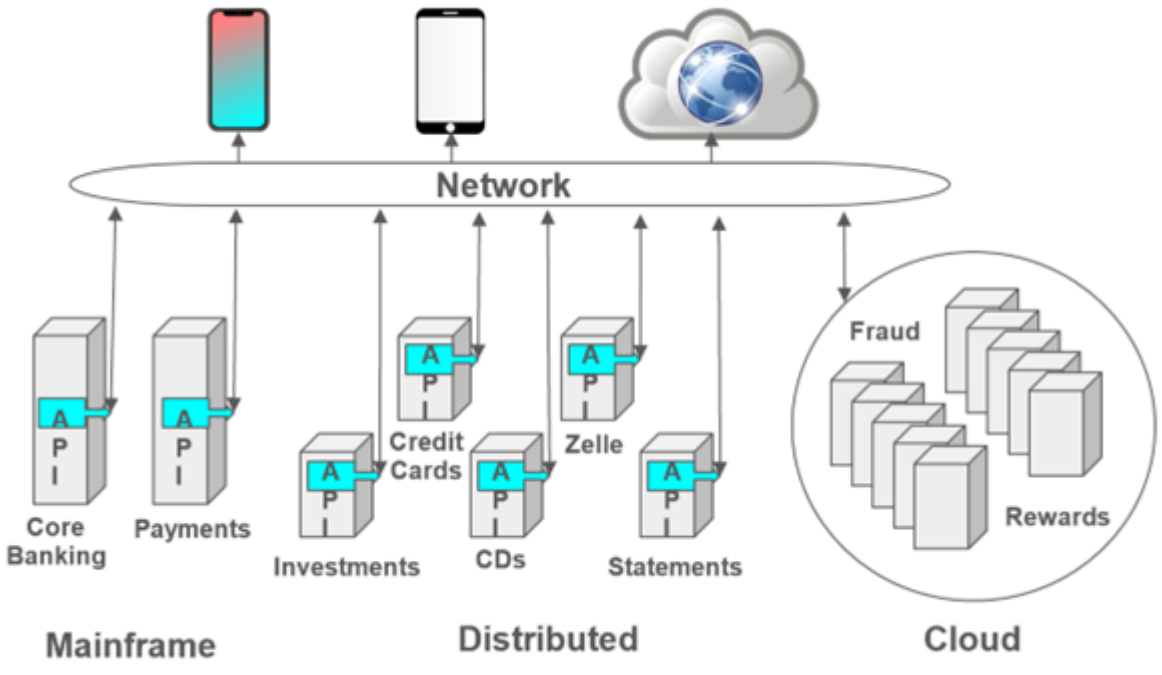

Figure 1: Sample banking application IT landscape

The interface-centric approach normalizes data access and sharing by transforming the data at the API integration points into a standard format.

This API standardization helps with reporting and analytics, providing consistent feeds for data warehouse applications and GenAI model training, and it lays an important foundation for innovation by simplifying the addition of new capabilities and applications.

Last but not least, it enables consolidation of functionality for web and mobile GUIs, such as displaying all bank accounts for a customer, or tracing the progress of an insurance claim across multiple systems.

Application API consumers and GUI users are not aware that an application is running on a mainframe, distributed system, or cloud.

This allows organizations to run applications on the right platforms for them, including modern mainframes, which are a good platform on which to run core transaction processing applications, as mentioned.

The Intellyx take

Mainframes have always been a cornerstone of commercial applications, and are actually more relevant than ever. This is especially true for the financial services applications that keep the world economy running.

Recent advances in mainframe hardware and software have brought mainframes into the 21st century, lowered cost, and reduced the impact of mainframe skill shortages..

New developer tools provide a common approach to mainframes, shared by cloud and distributed platforms. And AI-powered automation tools such as the BMC AMI suite simplify operations and data management — and mainframe automation for financial institutions extends this capability to business-critical workflows like core banking and payment processing.

Mainframe software providers such as BMC are also sponsoring programs to close the skills gap, offering developer training and methodologies for improving resiliency, performance, and security.

Mainframes remain the cost-efficient platform of choice for high volumes of transactions that absolutely must be performed accurately and stored and retrieved correctly.

It's an old adage that correcting an error is more expensive than preventing the error from occurring in the first place, but this is probably nowhere more true than in financial services. For many applications, mainframes are still the best solution to this cost effective calculation.

For more insight into the role the mainframe plays in financial services industry, as well as the effect of modern tools and processes, listen to the Modern Mainframe podcast, Banking on the Future: Tackling FinServ Challenges with a Modern Mainframe.