White Paper

Adaptive Application Modernization Strategies — Leveraging Resilient Practices, Platforms, and Hybrid Cloud for Business Innovation

Sponsored by: IBM

Melinda-Carol Ballou Peter Rutten
July 2021

IDC OPINION

Rapid, dramatic transitions to digitization moving into 2021–2022 drove and will continue to drive increased demand for effective application modernization for a range of existing systems on which businesses rely. Some examples are systems of record (SOR) and transactional and product management and development, enabled by adaptive process and automation strategies and resilient platforms. The majority of enterprises run key data on mainframes — successful modernization and digital innovation (DI) are enabled in part by leveraging current and innovative development techniques, including microservices, containerization and open source strategies, agile DevOps, and advanced analytics. These approaches continue to drive business responsiveness to dynamic change and can significantly improve delivery times, efficiency, relevance, and cost savings. They can also help transition staff and actively engage new, less experienced developers with software modernization (at a time of mounting demand for developer skills).

To enable DI and also to address increases in technical debt accrued with rapid shifts to both digital transformation (DX) and distributed hybrid work in 2020–2022, deliberate, incremental modernization strategies enable benefits of scale, including leverage of hybrid cloud, open source software (OSS) and standards, API strategies, and multilanguage support. IDC defines DX as the process of adopting "3rd Platform" technology (such as cloud, big data and smart analytics, and social/collaboration) and innovation accelerators (e.g., Internet of Things [IoT] and robotics) to evolve creativity, informed decision making, and DI to achieve differentiating speed, scale, and agility, helping create sustainable performance excellence in a digital economy. DI describes the ability for organizations to develop adaptive software to differentiate their businesses, enabling competitive advantage and disrupting the market. To create innovative digital products and services, enterprises must transform into software producers adopting new tools and technologies, embracing new strategies, and shifting their internal cultures. A critical need to transform with DX and to evolve adaptive software with DI impels application modernization and will continue to do so moving forward.

Heritage application modernization includes methods such as rehosting, replatforming, refactoring, integration with and leverage of modern development tools and approaches (including microservices and containerization), data strategies, API creation and management, and contextualized, appropriate platform choices. The performance, reliability, and security benefits of established high-end platforms such as IBM Z provide a foundational resource, especially for business-critical systems (such as

transaction processing [TP]). We recommend that organizations evaluate and leverage a platform's ongoing innovation for broader datacenter and cloud options to support expanded demands for DI.

That said, although these initiatives are urgent, disruptive "big bang" approaches can't work. Companies should establish incremental, planned evolution that encompasses organizational and process change, effective analysis and engagement of application portfolios, and appropriate automation strategies for intelligent analytics and execution.

IN THIS WHITE PAPER

This white paper discusses the imperative for mainframe application modernization strategies as a basis for digital transformation and digital innovation. Demand for coordination between key heritage systems that are foundational for businesses and modern development approaches enables DI via a combined strategy that includes process change, automation, and platform implementation. We examine this as an evolutionary, purposeful, and incremental process, along with the trends, benefits, and adoption opportunities based on IDC research. We define an end-to-end approach for application modernization options, including areas such as API integration and enablement, refactoring, rehosting, and replatforming, as well as integration with modern development environments and agile processes. Also, open source software technology, standards, and practices — along with modern development tools and rich platform support — provide a basis for consistent skills leverage across a broadening pool of developers, generalizing what historically required (and is still perceived as requiring) specialized skills. Challenges that must be addressed include the need for process and organizational change along with automation tools and appropriate platform adoption. We then consider IBM's broad portfolio including IBM Z, automation platforms, data management, intelligent analytics, and hybrid cloud strategy with Linux and the range of opportunities via IBM's relationship with and integration of Red Hat products such as OpenShift. (IBM's broad and deep services and partner offerings are also expansive resources.) We lay out user reference summaries for two IBM customers in banking. One is a major European banking service provider that diversified its mainframe systems to be able to move non-TP-oriented software to a distributed environment and microservices while leveraging IBM Z for TP-intensive, business-critical systems (using IBM, Red Hat, Linux, and IBM Z). This approach significantly cut costs and improved response and release times. Another IBM banking client with 35,000 employees and 8 million customers shifted to an agile DevOps approach for its mainframe systems, increasing its deployment velocity from two major annual releases per year to 24 deployment "go lives" per day (with 45 daily during peak times). This customer continues to rely on IBM Z for its transactional systems (while 70% of its noncritical applications are now in the cloud as part of this transition). We conclude with our perspective on future trends, challenges and opportunities, and advice about next steps.

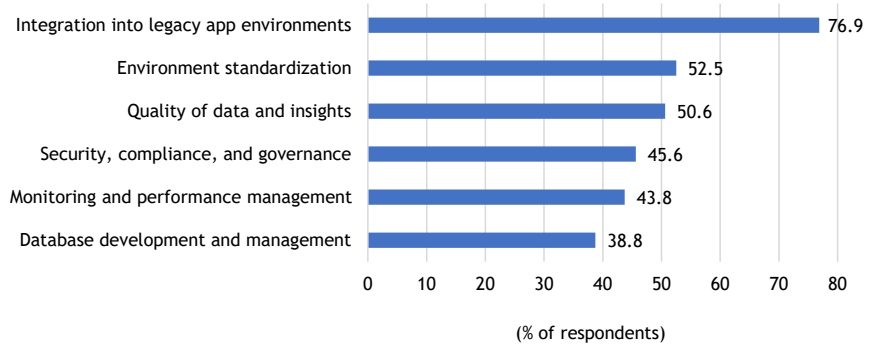
SITUATION OVERVIEW: MODERNIZATION STRATEGIES FOR DIGITAL INNOVATION

Making the transition to digitization enables not merely competitive positioning but business survival given a drastic rate of global change and complexity, which we expect to continue for the foreseeable future. The basis for DI for the majority of enterprises is grounded in existing applications, which demand modernization. IDC research shows that around 77% of organizations in 2020 found that integration with and modernization of legacy applications were major technology bottlenecks, followed by the need to integrate multiple environments and to address data quality and integrity (see Figure 1).

FIGURE 1

Top Bottlenecks to Application Delivery

Q. What are the top technology bottlenecks in your application delivery pipeline?



- Modernizing while dealing with legacy applications is a formidable challenge.
- Multiple environments and tools stacks can inhibit DevOps scalability.

n = 160

Source: IDC's *U.S. DevOps Survey,* September 2020

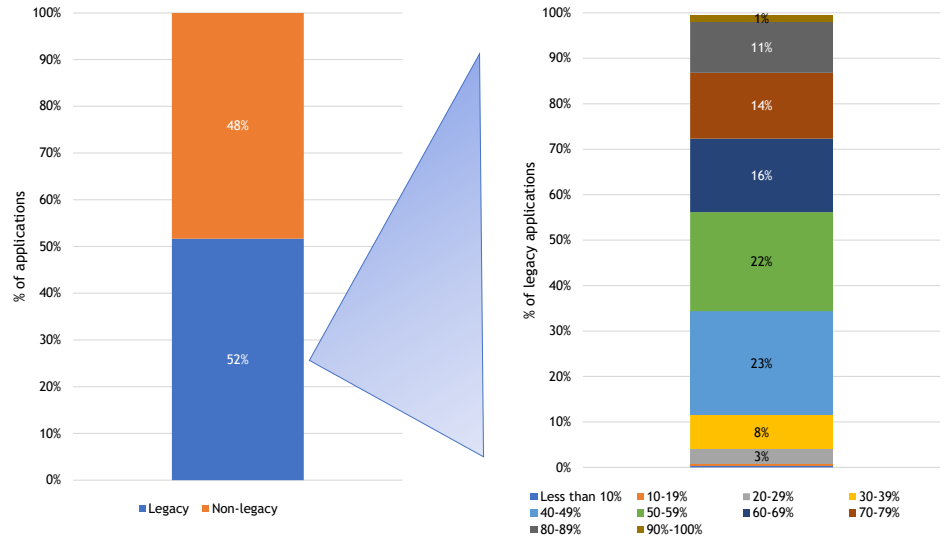
Exemplifying this drive to modernize, over the next two years, 80% of organizations expect to have modernized more than 50% of their applications, according to IDC's 2020 *PaaSView and the Developer* research. This means that addressing the remaining applications will require both mature and innovative approaches, since those applications that are easier to modernize (the lower-hanging fruit) will already have been tackled by many companies (see Figure 2).

FIGURE 2

Commitment to Modernization

Q. What percentage of applications in your organization are legacy versus non-legacy?

Q. Now — What proportion of your legacy applications have been modernized?



- 52% of applications within the enterprise are legacy applications.
- 65% of respondents noted that greater than 50% of their applications have been modernized.
- Enterprises have invested in modernization initiatives but still have much work to be done with respect to maintaining the technical debt specific to their digital estate.

n = 1,873

Source: IDC's *PaaSView and the Developer,* 2020

There is a pressing need in this context for a strategic perspective that encompasses efficient, evolutionary process and organizational transitions and change. Doing this in conjunction with automation adoption for modern application development, leveraging both emerging and existing enterprise platforms to incorporate business responsiveness, can help focus cultural change to adaptive processes. The use of open source strategies and standards and a range of commonly used development languages also provide a common basis for engagement in application modernization by generalized developer resource pools. With developer demand increasing and the average age of developers doing mainframe work decreasing, this generalization of skill sets is vital. These combined capabilities where modern tooling benefits from leveraging modernization techniques and agile processes and collaboration can help address skill shortages for application modernization.

IDC typically sees common strategies for application modernization: rehosting, replatforming, refactoring, and integration with "modern" development tools such as agile DevOps pipelines, integrated development environments (IDEs), continuous testing (CT), code analytics, and software composition analysis. Data strategies play a key role as well, including change data capture (CDC), data virtualization, event-driven strategies, and pattern analysis for complex data visibility, management, and refactoring. IDC research in 2020 showed that around half of developers have rehosted (51%) and 45% have replatformed and 44% have refactored an application in the past year. This research also indicated that 65% of respondents have modernized 50% of their application portfolio, exemplifying engagement and the need to further augment strategies to address the complexities of the remaining portfolio (refer back to Figure 2).

We also see hybrid approaches to agile adoption for software development and business engagement complementing application modernization (evolving traditional waterfall approaches). Often, popular views of agile development tend to be focused on fast, one-off creation of new applications. Yet iterative DevOps strategies are also key for modernization (along with incorporating integration and API initiatives), bringing in access to core heritage business applications, real-time data and services in systems of record, and other applications and data on mainframe environments to enable flexible business execution. Adaptive, judicious, and thoughtful approaches to the application portfolio and platform retention can help create efficiencies of scale for modernization.

Modernizing "In Place" for Efficiency and Platform Benefits

In evolving digital innovation across platforms, it can make sense to leverage transactions and data on the mainframe and not migrate. Choice of platform in that context is key. Depending on circumstances and context, one way of modernizing development for those systems involves keeping applications on the mainframe for security, performance, and viability and for currency and quality of service. This approach can also let developers without mainframe skills access traditional mainframe transactions and data "in place," which can be easier than exporting and transforming that data. Using REST-based APIs and tooling provides a way to directly access the data to support business and technology velocity.

For applications demanding high performance, security, and resilience, from an analytics perspective, there is value and often a necessity in leaving the processing on the mainframe. Benefits include the ability to perform in-transaction analytics and lookup to check for fraud during a transaction, giving the value of efficiency and consistency via the common platform.

Intuitive products that bridge the cultural gap between younger and traditional developers can help as well in transitioning organizations. Open source standards and approaches also are key factors in generalizing skill sets, providing a common basis for developers to engage.

The ability to coordinate DevOps initiatives across disparate teams, languages, control planes, and approaches, including cloud-based and hybrid cloud strategies (for either rehosting, replatforming, and/or refactoring), benefits from effective automation, platform solutions, and broad data access. Transforming software capability to be able to respond quickly to emerging business opportunities and modernizing processes to meet the needs of DI and emerging technologies can drive competitive positioning and success. Taking back-end systems and making data accessible can help achieve continuous delivery across multiple platforms and enable multispeed development teams to be able to benefit from existing information. (That data can otherwise be inaccessible and locked up.)

Technology Choices — Coordinating with IDEs, DevOps, and Platform Capabilities

Organizations should evaluate portfolio support for mainframe applications and practitioners and developers for modern development strategies. This should include effective integrated development environments and broad DevOps pipeline capabilities from design and software initiation with value stream management, through continuous testing, continuous integration (CI), continuous release (CR) and continuous delivery (CD), with code analytics and security throughout the process. Planning and prioritization applies here as well. Intelligent analytics with machine learning (ML) and artificial intelligence (AI) play a role increasingly for mainframe application transitions and as part of impact analysis and prioritization of modernization initiatives. Consider visibility into the application portfolios, impact analysis, and evolving capabilities for predictive analytics.

Platforms should be viewed and assessed dynamically and incorporate rapid innovation that improves upon core features with new generations, including extending systems to encompass cloud and hybrid cloud environments and effective coordination with application modernization environments and automation.

Process Issues — Evolving Adaptive Mainframe Development

We see a necessary shift to faster cycle times that involve a transition for mainframe developers. Instead of being siloed, with back-end organizations that change processes and releases on longer time frames (of six months or more), DI development cycles may need to occur daily, weekly, or even with hourly time frames. DevOps strategies incorporate coherent "end to end" life-cycle adoption on the part of development, operations, and business teams typically leveraging adaptive, lean approaches to code creation, testing, and deployment with faster, iterative cycle times.

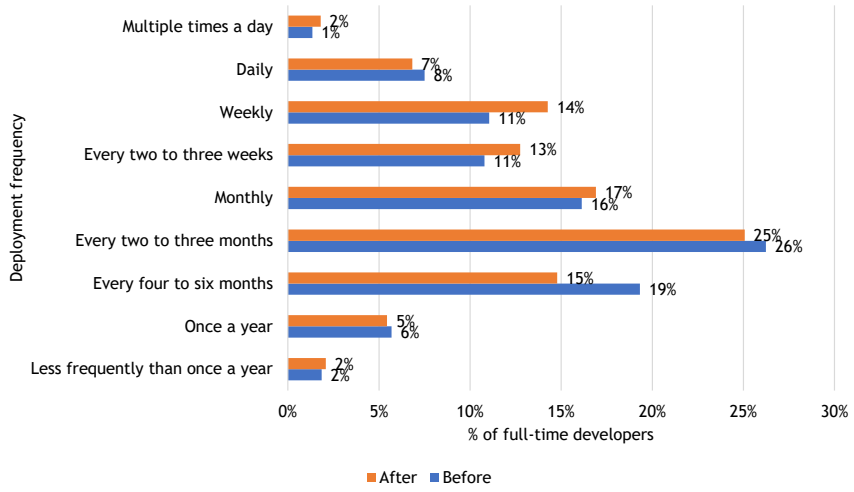
The ability to bring mainframe and distributed app teams together to coordinate with one another via end-to-end pipeline visibility can bridge that gap and help achieve delivery across multispeed teams and processes. This can let each team move closer to the velocity that the business requires. IDC research shows significant increases in shorter deployment time with DevOps adoption (see Figure 3).

FIGURE 3

DevOps Increases Deployment Velocity

Q. Prior to implementing DevOps, how often, on average, did your organization deploy code releases?

Q. After implementing DevOps, how often, on average, does your organization deploy code releases?



- DevOps continues to make development go faster.
- The following deployment cadences increased after the adoption of DevOps:
  - Multiple times a day
  - Weekly
  - Every two to three weeks
  - Monthly
- Deployments every four to six month decreased from 20% to 15% subsequent to the implementation of DevOps.

n = 741

Source: IDC's *PaaSView and the Developer,* 2020

Business users, analysts, and development and operations staff must communicate effectively about, and prioritize the evolution of, back-end systems to target other platforms and emerging technologies (like augment reality [AR] and virtual reality [VR]). This should include business-focused, effective design strategies to sustain corporate initiatives, such as making SORs available to front-end systems, as one example. The role of microservices and containers to democratize development for business stakeholders and others opens up opportunities for coordination, as well as use of open source approaches. The cultural divide across groups and ineffective common communication vehicles mean that back-end resources aren't leveraged and deployment can be difficult. Bridging those gaps via collaboration, automation, standards, a rich set of development languages, and process change is a key opportunity.

Make proactive decisions about incremental evolution of appropriate systems and prioritize those choices coherently (as opposed to reacting in ad hoc ways and deploying resources to execute in areas of less relevance to business needs). Even where value streams and/or requirements are communicated up front, business change and the lack of a common information-sharing vehicle across groups can still create misunderstanding and poor iterative development approaches. Effective coordination and application portfolio management can benefit quality control to improve relevant applications and to prioritize IT resources for iterative development and deployment of software that leverages vital back-end applications.

Businesses must also manage quality, testing, and security while responding to the exploding application content they quickly deliver (while containing high costs in a volatile worldwide economy).

TARGETING MODERNIZATION WITH IBM PRODUCT OFFERINGS

IBM has a differentiated, deep, and wide range of products and processes to modernize and to transition mainframe applications and environments with its portfolio. The breadth of IBM's capabilities encompasses software inception, integration, development, delivery, and data management on a foundational platform. Examples range from core and critical runtimes (with CICS and IMS) to compilers and language support (from COBOL and Java to Node.js, Python, and Go) to data management including CDC, data virtualization, event-driven strategies, and pattern analysis for complex data visibility, management, and refactoring (with Db2 and other tools such as command and query responsibility segregation [CQRS]). IBM also supports API creation and management and z/OS Connect EE; provides developer support with IBM Wazi Developer, Red Hat OpenShift, and Ansible; and offers analytics with Application Discovery and Delivery Intelligence (ADDI) and ZD&T and the Z platform itself. This functionality is interwoven; synergies across these and other aspects of the IBM portfolio can enable supple and resilient approaches to modernization. Organizations require approaches including hybrid cloud, microservices, and containerization and support for process change, which IBM offers.

Strategies to modernize z/OS applications and workloads include:

- Analyzing the application portfolio
- Defining interactions between applications as APIs — containerizing and benefiting from microservices (as an example)
- Shifting individual services into contemporary languages and running them on z/OS or Linux on IBM Z

- Integrating z/OS into a common open hybrid cloud platform based on Red Hat OpenShift across the enterprise and applying cloud-native tools and practices to z/OS

Gaining visibility into application portfolios is a foundation for modernization, and IBM's Application Discovery and Delivery Intelligence enables application analytics and predictive, actionable context. ADDI's functionality lets development teams assess application interdependencies, complexity, and quality across platforms, environments, and languages. This can enhance application understanding and decision making for organizations and developers to modernize core business assets more easily and for broader platform support (identifying API candidates, discovering business rules, and providing impact analysis, as examples).

IBM's overall portfolio lets organizations set objectives and contextualize applications for appropriate workload fit. Incremental, judicious, and efficient transitions for modernization are facilitated by IBM's Z platform and surrounding automation capabilities (along with needed process and organizational change). This includes the ability to both modernize assets and deliver new application services with Z and LinuxONE for public and private clouds.

A goal of IBM and IBM Z is to provide flexibility with hybrid cloud support — including defining application interactions as APIs; leveraging adaptive modern languages such as Node.js, Python, Go and, of course, Java; and being able to run on z/OS or Linux on IBM Z and apply native cloud practices and solutions on Z.

The performance, reliability, and security benefits of IBM Z as an established high-end platform are a core resource for many organizations, especially for vital demands like business-critical transaction processing. IBM Z is being innovated on a continuous cadence, bringing ongoing benefits to customers. The ability to build hybrid cloud infrastructure with IBM Z enables flexibility and resilience for organizations, helping vitalize their application modernization efforts with appropriate infrastructure support.

Support for Red Hat OpenShift is a key capability of IBM Z as containers and OpenShift abstract underlying differences between IBM Z and other platforms. This lets developers use standard, broader skills; modern toolkits; and open source strategies while working on IBM Z. IDC sees containerization spreading to mission-critical and security-sensitive workloads that typically run and rely on IBM Z, facilitated by these capabilities. Multimodal delivery and engagement are key benefits — IBM Z plus Red Hat provide modernization and new application services capabilities to create native participation in that ecosystem. This boils down to enabling customer controls in the Kubernetes control frame — being able to leverage the Kubernetes controller to interact with traditional services and runtimes and driving the ability for users to transition over time while being able to create net-new services.

Red Hat OpenShift's cross-cloud, broad OSS developer base and coordination with IBM products enable execution across environments for modernization. Examples of the combined portfolio include z/OS Cloud Broker V1, integration of IBM z/OS into OpenShift Container Platform, and z/OS Cloud Broker V2, z/OS integration with OpenShift via Ansible for configuration management and orchestration application delivery using Red Hat Ansible Certified Content for System Z.

Untapped Red Hat Ansible opportunities remain on the deployment side, including coordination and synergies with IBM's DevOps Engineering Workflow Management (EWM) and UrbanCode Release and UrbanCode Deploy and distributed testing. That said, RHEL, KVM, and Open Shift Container Platform (OCP) stack on Z is strategic for Red Hat/IBM and improved support and integration across the portfolios emerging.

Open Source Software and standards are a backbone capability for IBM's and Red Hat's portfolio and enable broader, standardized adoption across developers, both in an x86 context for development and also on IBM Z. IBM Wazi Developer for Red Hat CodeReady Workspaces enables developers to use desktop IDEs like Visual Studio Code and Eclipse, or a containerized web-based IDE such as Eclipse Che. Users can leverage a containerized, z/OS sandbox for development and test on Red Hat OpenShift running on x86 hardware, which can then be deployed into production on native z/OS running on IBM Z. IBM's evolving and deepening partnership with GitLab enables close integration for CI/CD, and additional capabilities in that context are emerging. IBM also enables agnostic cloud-native development by using standardized tools like Git and Jenkins for parallel development and a standard CI/CD pipeline while staging deployment on popular artifact repos like Artifactory or Nexus. And to address quality issues, IBM's Virtual Test Platform enables developers to shift left with transaction-level testing.

IBM Cloud Paks can help organizations build, modernize, and manage applications across cloud platforms (also encompassing security considerations). IBM Cloud Paks include pre-certified containerized software and services that can provide customers with a common operations and integration framework. Built on Red Hat OpenShift and IBM's hybrid cloud platform, their goal is to enable a consistent experience for an infrastructure management control plane that includes AI, automation, and security. IBM's hybrid cloud support enables companies to work across public and private clouds and traditional environments.

IBM's Garage method provides a clear, straightforward on-ramp for process change and scaffolding for application modernization (and other needed areas of innovation and transition). And IBM's services arm with GBS provides proactive transitional and longer-term support for modernization practices (as needed).

Additional differentiating resources include IBM Research Cloud Innovation Lab (CIL), giving access to IBM Research's expertise, skills, and technologies to help clients and partners leverage emerging cloud technologies. Lab members can interact and collaborate with IBM researchers, view demonstrations, and experiment with the most current OpenShift hybrid cloud platform innovations, for instance.

IBM's partnering and third-party integration strategy also brings flexible options for popular development and collaboration environments (including IBM's expanding GitLab partnership, Git and GitHub integration, CloudBees, and IDE support).

IBM's differentiation with the Z platform and ongoing investments provide a foundational basis for application modernization.

- IBM launched the latest generation of IBM Z, the IBM z15, in September 2019, and IDC expects the next release to be introduced in 2022. The current system is fitted on a standard 19in. frame, provides a 25% capacity increase, and features Privacy Passports, to help protect and secure mainframe data even as it moves off the platform to a distributed or cloud environment. Most importantly, the z15 is heavily focused on hybrid cloud and supports Red Hat OpenShift.
- The earlier IBM z14 included an embedded encryption engine to support data security with little performance overhead, and the z15 provides a similar embedded hardware solution for data compression with a 6:1 compression ratio. Compression on the chip improves the speed of processing the massive volumes of data that today's businesses require.

- IDC anticipates the need moving forward for expanded hybrid cloud capabilities, additional security features (to help defend against future, highly sophisticated attacks), platform flexibility to integrate with datacenter and cloud, and more robust AI inferencing capability to enable workloads to run AI models at high transactional and analytical processing speeds.

In summary, the overall breadth of these combined capabilities and platforms differentiates IBM in addressing application modernization demands — giving businesses the opportunity to leverage existing and new services and to bring increased flexibility, agility, efficiency, speedy delivery, and security as organizations transition their heritage systems to digitally transform.

CUSTOMER REFERENCE SUMMARIES

IDC spoke with two companies engaging in application modernization strategies leveraging IBM technology with salient use cases — one chose a "best fit" approach to leverage IBM Z for its business-critical, TP-intensive software while shifting its non-transaction-oriented, traditional z/OS applications to a distributed microservices environment using IBM's portfolio and Red Hat's offerings with OpenShift. Another modernized its mainframe development to an agile DevOps environment to move from two major annual software releases per year to 24 deployments per day, using IBM Z for transactional systems.

Bank Service Provider Leverages Hybrid Modernization Strategy

A major European service provider for cooperative banks needed to determine how to combine new and existing technologies and applications to be able to move back and forth smoothly across both. To be responsive to competitive demands and customer needs, it was vital for the company to be able to evolve and modernize while leveraging existing applications. With about 600 banks as members, 150,000+ workstations, 500,000+ virtual servers, 8 mainframe computers at approximately 250,000 MIPS, and 80+ billion mainframe transactions per year, the company's environment is massive (providing banking support for about one-third of the population of the country in which this organization resides). Many of the core applications that are enabled by the banking service provider run 24 x 7 and couldn't be disrupted, so the transition needed to be stepwise and gradual, rather than the risks of a big bang replacement project. Evolving the company also relied on a few key leaders who were able to bridge the gap with knowledge and experience across existing strategies and evolving open systems and microservices approaches.

The company has now decoupled its mainframe systems to move to a DevOps mixed, distributed microservices and mainframe environment leveraging capabilities from IBM, Red Hat, Linux, IBM Z, and related technology, significantly improving flexibility, efficiency, response, and release times.

The company's core banking applications run on IMS-TM and Db2 for z/OS (as its RDBMS), which the teams plan to continue to leverage. The company described this as "rock solid" technology combined with the characteristics of IBM Z to help guarantee data consistency, availability, and security, along with self-established processes to manage it all. From an applications perspective, the business logic is implemented as IMS transactions (which are consumed in either a synchronous way via IMS-Connect or asynchronously via IBM MQ). There is also a significant amount of batch processing.

In a pragmatic strategy, the teams had earlier introduced a mix of both COBOL and Java (15 years ago for batch processing and 10 years ago for IMS), which give them the ability to replace existing COBOL subroutines with Java without the need to rewrite large programs, to help provide smooth evolution.

Teams at the company are moving to containers in the open systems area, replacing monolithic server-based applications with a microservices approach. They are using Red Hat OpenShift as a platform for microservices development. Currently used in the x86 environment, they will be able to switch on OCP Linux on IBM Z to get the benefits from colocation to avoid network communications issues, leverage new patterns and modern CI/CD pipelines, and take advantage of IBM Z for high performance, security, and more efficient, cost-effective execution.

Since the company had followed a service paradigm for around 30 years (where the IMS transactions were more or less comparable to a REST service without 3270s interacting with IMS), this also positioned the company's teams to reengineer the middle tier of the company's three-tier application by replacing it with an API layer. They were able to reengineer the monolith, cut it down to domains (with bounded context), and reimplement it in a microservices architecture (the technology layer is OCP). The company's objective is to move the midtier layer to OCP and to move away from proprietary frameworks and datacenter management tools. So while they are reengineering the back end (IMS-TX/Db2) according to the microservices paradigm, they are still using and leveraging classic technologies and incrementally extending those with new technology.

They are expanding their z/OS environment by introducing Linux on IBM Z on the same platform where z/OS resides, to benefit by colocating x86 apps, which intensively interact with z/OS applications. Physical network traffic is being replaced by in-memory networks. This generates significant performance benefits by eliminating network latency, which is valuable for online processing. In batch processing, the teams at the company improved processing time by a factor of nearly twice as fast to help guarantee customer service-level agreement (SLA) requirements. The overall benefit to the organization is maximizing throughput by minimizing resource consumption and helping eliminate network bottlenecks.

As an example, the company had a performance issue in the payment area for a client, where 2 million transactions/payments had to happen in a very short time frame; when the company first brought in the customer, these payments took three to four hours and it urgently had to find a way to process faster. By avoiding network latencies and database contentions with Java on z/OS, it was able to execute the 2 million transactions in less than an hour. It expects to extend this pattern to other applications and areas of the company.

Another challenge for the company was not technology related but resulting from inexperience and cultural resistance on the part of staff who narrowly focused on either x86 and open systems or IBM Z and mainframe without a broader perspective. Vital to its transition was a core set of leaders who had a hybrid perspective across both, and educating and mentoring teams across the divide. IBM's acquisition of Red Hat played into the company's strategy in this area — it had already opted for Red Hat pre-acquisition for its open source, cross-cloud, and microservices capabilities. IBM's combined strategy with IBM Z-related products and technologies and Red Hat enables a portfolio that is underpinning the company's application modernization strategy. (This company works closely with IBM to help drive solutions, such as the combined portfolio with Red Hat.)

The company's direction includes:

- Continuing to modernize applications using new technology and concepts without disruption, including leveraging new development languages with applications written in COBOL
- Contextualizing decisions based on business needs and the scope of required application changes to either retain them in IMS/COBOL or implement as a microservice (storing the data

on Db2/Z means the data can be accessed from both the new and the old worlds to cover the coexistence phase)

- Establishing in the next 12–18 months a full development and production environment for the bank's combined strategy

The company advises peer organizations to start modernization efforts with small projects that are meaningful to the business — reengineer incrementally, using staff who know how the existing application works and who are also able to bring in new approaches (supplementing with external mentoring and services as needed). Leaders at the organization recommend building on initial successes to grow the modernization initiative.

Bank Dynamically Increases Deployment Speed and Resilience as It Re-Architects Application Portfolio Leveraging IBM Z and Cloud Options

A major bank and global leader in the niche food production, factory, and logistics chains arena, with 35,000 employees and 8 million customers, reached a point where it needed to modernize its application portfolio, including core transaction processing systems. The company shifted to an agile DevOps approach for its overall software portfolio, including its core payment/transactional platform built in COBOL running on IBM Z. The company has 12,000 staff in IT and 3,000 dedicated staff to support the company's TP system. 30 years of history were embodied in the application at the time, and 25 businesses would have to wait for one another whenever functional changes were needed. Because of painstaking, infrequent, expensive, complex, and risky deployments, the teams decided 10 years ago to double down on their COBOL investment for the entire payment system on IBM Z platform and establish a DevOps strategy. (This was a fundamentally different approach than treating these COBOL applications as "legacy" — the company relies on this strategic TP capability.) The company had merely two high-risk releases per year at the time, with costly change management and testing demands, and for that reason, it chose to re-architect.

To create efficiency, the teams separated out the payment-specific aspects of the TP application and architected delineated, target applications for more complex related use cases that had previously been embedded. For instance, credit card applications that required complicated workflows (with 25+ additional verifications) were shifted to a credit card–specific application and simple payments were addressed in the TP system. While the volume of the core system decreased dramatically as a result of this approach, the TP application was still massive. So the organization worked in parallel to standardize development and automate its DevOps processes, testing, and CI/CD pipelines.

As a result of its transition, the company was able to increase deployment velocity from 2 major annual releases per year to 24 deployment "go lives" per day (with 45 daily during peak times). This customer continues to rely on IBM Z for its transactional systems (while all of its noncritical applications are now in the cloud as part of this ongoing transition).

The company's strategy involved a fundamental enterprise architecture redesign and decisions to analyze and determine which services were mission critical. The teams considered applications they are meshing with service architecture to move into the future — for instance, rebuilding mono apps that were big bang originally to be able to move to a clean service-oriented architecture (SOA). This approach took two years with some 50–60 architects working in that area before writing code; when moving toward DevOps, the company changed its structure, fundamentally reorganizing its IT and business organizations from the CEO/CIO level down to have both groups working together collaboratively with joint goals and key performance indicators (KPIs) and a common suite of tools. The company took a holistic architectural approach — considering infrastructure and business

processes, analyzing its portfolio, and standardizing on a DevOps tool suite. Tools included the IDz workbench, ADDI for impact analysis, SonarSource, service virtualization, Engineering Workflow Management (EWM, formally Rational Team Concert [RTC]), and UrbanCode Deploy for deployments. The teams found that this was successful functionally. But deployment and centralized management across the tools was challenging (as they needed to incorporate CICS, Db2, and COBOL).

The company's intent is to completely move to the cloud and away from on premises by 2023 with only one exception — the Z platform. For many reasons, its Z COBOL applications are so tied together with the capabilities and benefits of the platform that they are not portable. The company sees IBM Z as by far the fastest and most secure option, and for that reason, the Z platform will stay as it is. Every application that has combined requirements to be highly scalable and highly secure and with high data integrity requirements must run on IBM Z. (Applications that don't have that triple application requirement will run on the cloud — on Linux on IBM Z, AWS, or Azure — and not on Z.) Even payment configuration services, for products that have specialized portals configured, run in the cloud while TP and high-volume systems run on IBM Z. (And they are not containerizing these applications, which have 2 million to 5 million lines of code.)

Also, OSS products are becoming more powerful and more well integrated and moving at a speed where a closed source vendor will have a hard time keeping up with that speed. OSS is more and more becoming the dominant streaming and tooling option for this organization.

This company's approach has been long term, beginning with the strategy stage re-architecting the TP application. Teams pursued this on two tracks, after two years of strategy; one track worked on rebuilding applications and the other track worked on reevaluating the toolbox. The shift to the EMW (formerly RTC) enabled a fully automated CI/CD process. With a move toward Azure DevOps for orchestrating the end-to-end process, the teams are evolving their pipeline. In parallel, the redevelopment of core business apps is ongoing. The teams described this overall process as "open heart surgery while a patient is running a marathon."

In advising other companies, the leaders recommend having a service architecture, and to make conscious choices to use the mainframe appropriately (where high-end requirements demand it), while putting other software on the cloud for appropriate cloud leverage. They also found that CI/CD standardization is key, along with executive buy-in and evangelism for success. And the CI/CD pipeline also has to be flexible, adaptive, and designed for change. The leaders of the process also recommend an incremental, well-planned process — "don't try to win the entire world in one day," establish the process first and then enrich it with additional controls. They recommend being aspirational but not to make choices further out than two years, understanding this process as an evolutionary journey rather than something that users can predict that far out in time. That said, DevOps resilience is a benefit. What surprised leaders with regard to COVID-19 impact was that their development and deployment velocity was able to remain high despite challenges (though they did have issues with sites becoming less stable, which they were able to address).

FUTURE OUTLOOK

We expect additional evolution of mainframe modernization systems in the areas of intelligent analytics. Increasingly, organizations seek proactive surfacing of data to enable pragmatic decision making with actionable insight (rather than being locked up invisibly in automation systems) and evolving predictive analytics capabilities. We see democratization of development with microservices, containerization, ML, AI, and emerging robotic process automation (RPA) as key aspects for business

engagement and relevance and to help address looming developer shortages. Coordination across business and technical teams on the mainframe side with collaborative capabilities is also key, including executive stakeholders, architecture/design, quality, and security, to deploy with greater resilience and improved performance. This should incorporate planning for appropriate prioritization and audit as part of change and configuration management (for compliance).

CHALLENGES/OPPORTUNITIES

Shifting Cultures for Digital Innovation

Digitization is not optional. Businesses thrive by leveraging DI with software delivery as a foundation for execution. Effective application modernization revitalizes organizations and is a lynchpin for successful execution in an unpredictable, dramatically complex global environment. Businesses must be able to be adaptive across wildly varying needs — to be dynamically responsive to changing customer, business, compliance, risk, and value stream requirements; incorporate corporate responsibility for sustainability and diversity; and be flexible for whatever happens next (that we can't yet predict). Yet change — and process change especially — is tough. Human beings are wired for consistency more than we are for change.

Establishing a successful and secure modernization life cycle as a link between development and deployment requires cultural and organizational change, leveraging what has worked well previously to transition. Executive leadership buy-in and support are core. Staff across business, development, and operations teams need to understand evolutionary phases for modernization. The staff must engage in necessary analysis of functional capabilities required to incrementally transition organizations to enable back-end mainframe systems' coordination with new applications and strategies. The range of needed solutions in this context include development for mainframe systems and high-performing platforms that are accessible to and usable by a new generation of developers, including microservices development, containerization, quality tools/continuous testing, code analytics and security, data management, API support and management, and release orchestration. Leveraging more closely linked processes and phases and required functional and technical capabilities as well as process, tools, and organizational needs to create an effective evolution from back-end mainframe to containers and microservices to DevOps can be key for a successful transition.

Approach application modernization in an evolutionary manner to benefit from access to and coordination between existing, real-time data systems, with the velocity and business visibility enabled by microservices. Also, as embedded software feeds competitive product positioning, we see the expansion of delivery platforms to products and systems of systems or the Internet of Things and the edge. These may often also require coordination with existing SOR on the mainframe. IDC expects that the levels of complexity demanded by these systems will drive cognitive use by developers both in their applications and to support development. (IDC expects that by 2023, 66% of all applications will be beyond piloting machine learning and artificial intelligence as part of development and DevOps.) Iterative approaches to development and deployment and leverage of back-end data existing on this wide variety of emerging, multimodal platforms, in conjunction with legacy platforms, are playing a role now and will increasingly do so, driving successful digital innovation strategy.

The greatest challenges for most organizations in shifting to digital innovation are the necessary cultural and organizational transitions. Nonetheless, benefits are pushing established mainframe organizations to shift to adaptive processes and tools, toward connected teams, business stakeholders, and modern development approaches. The 2020–2021 pandemic exemplified the ways

in which human beings can react to dramatic, unexpected change adaptively and creatively — often heroically. That said, the tendency for developers that have long-term, established approaches is that those habits tend to be entrenched and harder to shift. These kinds of barriers must be addressed directly with executive leadership and evangelism and an implacable and significant commitment to training, process, and organizational change; invest in appropriate automated technologies, and as needed, bring in resources to mentor and feed change.

CONCLUSION

Organizations should evaluate their current, existing application portfolios to assess required effort to modernize apps and appropriate strategies. Prioritize based on key value propositions and business criticality, and adopt efficient approaches. Gain executive leadership buy-in and support to create a secure modernization strategy that coordinates development and deployment along with organizational and process change. Start with small projects that are also meaningful to the business — reengineer incrementally, using those who know how the existing application works and who are also able to bring in new approaches (supplementing with external mentoring as needed). Build on initial successes to grow modernization initiatives. Set early and continuous quality and security management — identify modernization metrics/KPIs and milestones to help determine progress for software assets. Be pragmatic in transitioning. Establish proactive design and quality management standards with analytics across apps for pragmatic agile DevOps life cycle that encompass resilient, adaptive, and secure multimodal platforms.

About IDC

International Data Corporation (IDC) is the premier global provider of market intelligence, advisory services, and events for the information technology, telecommunications and consumer technology markets. IDC helps IT professionals, business executives, and the investment community make fact-based decisions on technology purchases and business strategy. More than 1,100 IDC analysts provide global, regional, and local expertise on technology and industry opportunities and trends in over 110 countries worldwide. For 50 years, IDC has provided strategic insights to help our clients achieve their key business objectives. IDC is a subsidiary of IDG, the world's leading technology media, research, and events company.

Global Headquarters

140 Kendrick Street
Building B
Needham, MA 02494
USA
508.872.8200
Twitter: @IDC
blogs.idc.com
www.idc.com