Rocket® software

From monoliths to microservices: Four IBM® i modernization best practices

Contents

03 Introduction

04 Best practice #1: Align IT with business

05 Best practice #2: Choose where to deploy workloads

06 Best practice #3: Break monolithic applications into manageable chunks

07 Best practice #4: Build a consistent data-driven plan

09 Get started on your modernization journey

Introduction

In today's disruptive world, many organizations are shifting application modernization from monoliths to API/REST services and eventually to microservices via refactoring. Thanks to advances in cloud technology, the demand for microservices is increasing at a large scale. Unlike monolithic systems, microservices are designed to be able to scale along with changing market demands. As such, modern enterprises are moving towards microservices to stay ahead of the curve.

These digital capabilities are more than just hype—they deliver real business value. For example, Netflix's move from a monolithic to a cloud-based microservices architecture has made the company more scalable and efficient. Today, Netflix runs 24/7, is scaled to the next order of magnitude, and is optimized for speed. As business leaders see successes like this, they, too, demand similar dramatic transformations from their IT departments.

The biggest consideration for anyone tasked with driving major transformation is how best to implement it to deliver the most value for the company. This eBook sets out four modernization best practices to help evolve your IBM® i applications from monoliths to APIs and cloud-based microservices. But, before that, let's first understand the difference between monoliths, APIs, microservices, and refactoring.

- Monolithic applications are large applications. They are typically complex and designed to handle multiple related tasks. They encompass several tightly coupled functions, and given their broad scope, they tend to have massive code bases.
- Application Programming Interfaces (APIs) are a set of definitions and protocols to build and integrate application software. APIs list several operations that developers can use, along with a description of what they do. Developers don't necessarily need to know how, for example, an operating system builds and presents a "Save As" dialog box. They need to know that it's available for use in their app.
- Microservices arrange an application as a collection of loosely coupled services. The goal is to reduce development complexity, ease deployment headaches, reinforce resiliency, and increase release velocity. In a microservices architecture, services are fine-grained, and the protocols are lightweight, making this architecture ideal for cloud and edge deployments.
- Refactoring involves restructuring the source code of an application or piece of software to improve operation without altering functionality. Developers review the software architecture and, where possible, refactor code to create smaller and more manageable components or microservices.

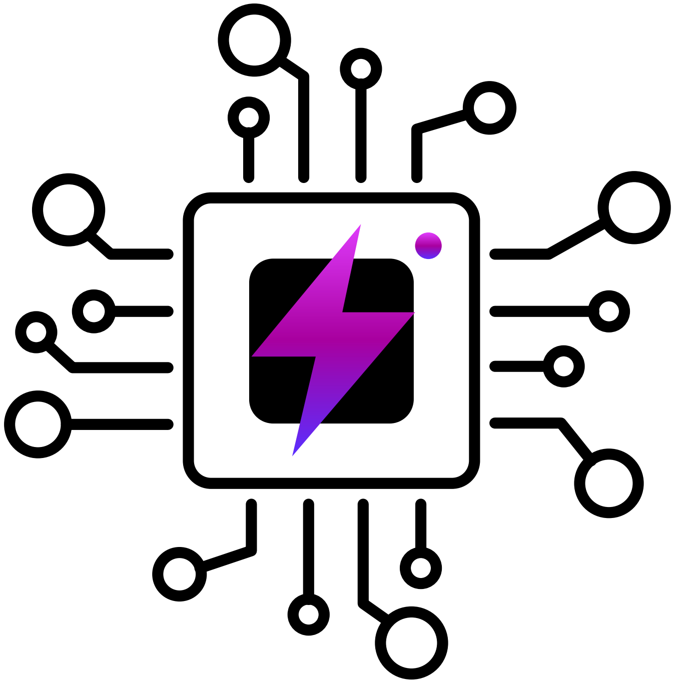
Now, let's dive in.

Best practice #1: Align IT with business

When it comes to modernization, one of the most persistent challenges is aligning IT with overall business objectives. All too often, IT has a good idea of where to start in terms of breaking down monolithic applications and deploying them in the cloud, but they may not have the complete picture when it comes to how this will transform the business. This is partly because IT departments often work in silos within organizations and don't have visibility into the organization's strategic plans.

Moreover, IT teams traditionally take a bottom-up approach, asking how they can move monolithic applications to the cloud to save costs. On the other hand, business leaders take a broader top-down view and ask how they can improve team productivity and better respond to market and customer needs. The disconnect between these two perspectives causes the IT-business gap to persist.

To overcome this challenge, it's vital for business and IT to collaborate from the get-go. This will help avoid piecemeal fixes and ensure a more holistic, strategic approach. By working together, business and IT can ensure that the modernization process aligns with the organization's overall objectives, resulting in a more successful transformation.



Best practice #2: Choose where to deploy workloads

If you run IBM i, a key part of modernization is understanding what type of IBM i user you are. Is IBM i your only IT infrastructure, or is it part of a larger IT landscape that includes other systems such as Linux and Windows servers? Moving from IBM i to IBM + other systems is a journey most businesses take. To figure out where you are on this path, consider what you rely on today, what you will need in five years, and which business drivers are influencing your current and future environment.

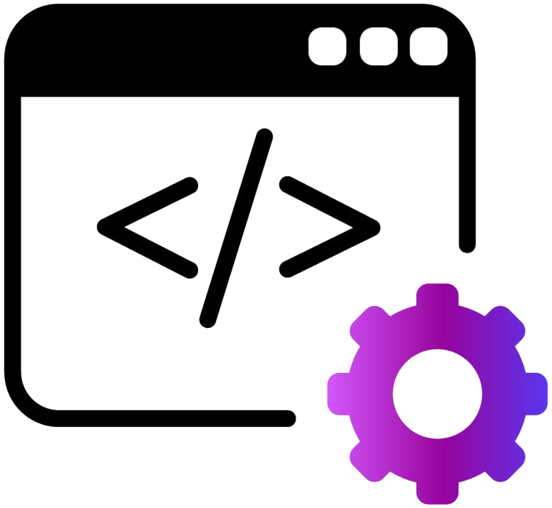
Once you know what type of IBM i user you are, you can start to think about how to break up monolith applications and how IT can leverage cloud technologies. For example, if your front-end user interfaces combine RPG applications with Python, Java, or another type of code, you've entered the world of multi-code languages and applications. Your environment has become more complex, but you have opened the door to more options to serve the business and your customers. This information can help you take the first step in the journey and begin to explore your cloud options.

There are multiple options for implementing a cloud strategy, and the best way to proceed is to design a plan that optimizes business outcomes. Consider which transactions and workflows truly need the capabilities of an on-premises IBM system and which could be moved to the cloud. Moving to the cloud doesn't have to mean moving everything to the cloud—it is possible to take a blended approach with some applications living as microservices or partially in the cloud and on-premises. It all depends on the specific use case.

Best practice #3: Break monolithic applications into manageable chunks

Monolithic applications can be thought of as the giant “boulders” of the software world. They are tough to rewrite, so much so that many organizations decide to scrap them and start writing new software from scratch. Instead of undergoing that labor-intensive, time-consuming, and expensive endeavor, advanced tools can be used to take a tactical approach to turn applications into APIs based on user needs, business needs, and budgets.

Turning monoliths into APIs doesn’t have to be done simultaneously. Areas used most in the application can be turned into APIs consumed in mobile applications, websites, etc. This way, the “boulder” can be chipped away at, and progress toward modernization can be made based on informed priorities. For example, when you turn your monolithic applications into APIs, you can build a new user interface while retaining your existing back end. When the time is right, you could begin modernizing your back-end software using the cloud and a different database without having to make changes to the front end again.



By breaking down a monolithic application into manageable chunks, the process of modernizing can be more strategic and less daunting. Focusing on areas that will make the most significant difference to your business is also essential. The best way to do this is to surface the end-to-end workflows for your legacy applications and give IT and business visibility into how people use them. This way, you can prioritize the most urgent needs and move on without relying on opinions or guesswork. Adopting these tactics allows applications to be updated incrementally, which is less disruptive and can be done more quickly and affordably than a complete rewrite.

Best practice #4: Build a consistent data-driven plan

When modernizing or automating legacy applications, it is best to take the “Intelligent Legacy Automation (ILA)” approach. This comprehensive and data-driven strategy includes six crucial steps:

01

Process discovery

Understanding how your company uses IBM i applications before beginning modernization or automation efforts is vital. This will help you properly assess the time and resources required for the project and avoid potential cost overruns. Relying solely on input from the business can lead to inaccurate estimates, so it is essential to take a data-driven approach that includes both business and IT stakeholders.

02

Project prioritization

By gaining visibility into your workflow and processes, you can identify areas where automation and modernization will impact the business most. This ability to prioritize enables you to set the project up for success and gain approval from vital business stakeholders. You can also show a direct link between the proposed work and the business impact.

03

Automation execution

Once you have a clear plan, the quickest way to see ROI is to start with RPA. Building robots to take on mundane tasks within a workflow can improve productivity. This frees users to focus on more exciting work, increasing user satisfaction.

04

Modern user experience

When looking at workflows that involve navigation across multiple screens, consider areas that slow down the user. However, rather than simply building a modern UI that mirrors the green screen interface, it is more effective to build a UI that fits the user’s workflow. This can involve consolidating screens and rearranging fields to create a more intuitive flow.

05

Process integration

Building APIs to integrate your IBM i applications with other critical business applications can improve business alignment. Process integration can “un-silo” legacy applications, creating a system that allows business operations and customer engagement to be less constrained by IT infrastructure.

After publishing an API, you must determine the most efficient ways to implement it. One option could be to bypass the terminal screen and directly invoke the application’s business logic. Another could be to try and access data in the application’s storage system. Response times are critical, so if possible, investigate these options to speed up the execution of the API. More information can be found here.

06

Automation management

Once you’ve built robots and APIs, you can further improve their development and deployment using a DevOps-lite tool. This will help to automate processes, enforce policies, and improve developer productivity. By doing this, you will be able to modernize and automate your development environment more effectively. Modernization—including automation—is not a “one and done” journey. Businesses often find themselves making ongoing changes. The ILA process can be repeated each time there is a change in platform, business, or customer request. Whether moving to a new CRM system, a bottleneck in the call center, or other issues, revisiting this process can help pinpoint and fix the problem.

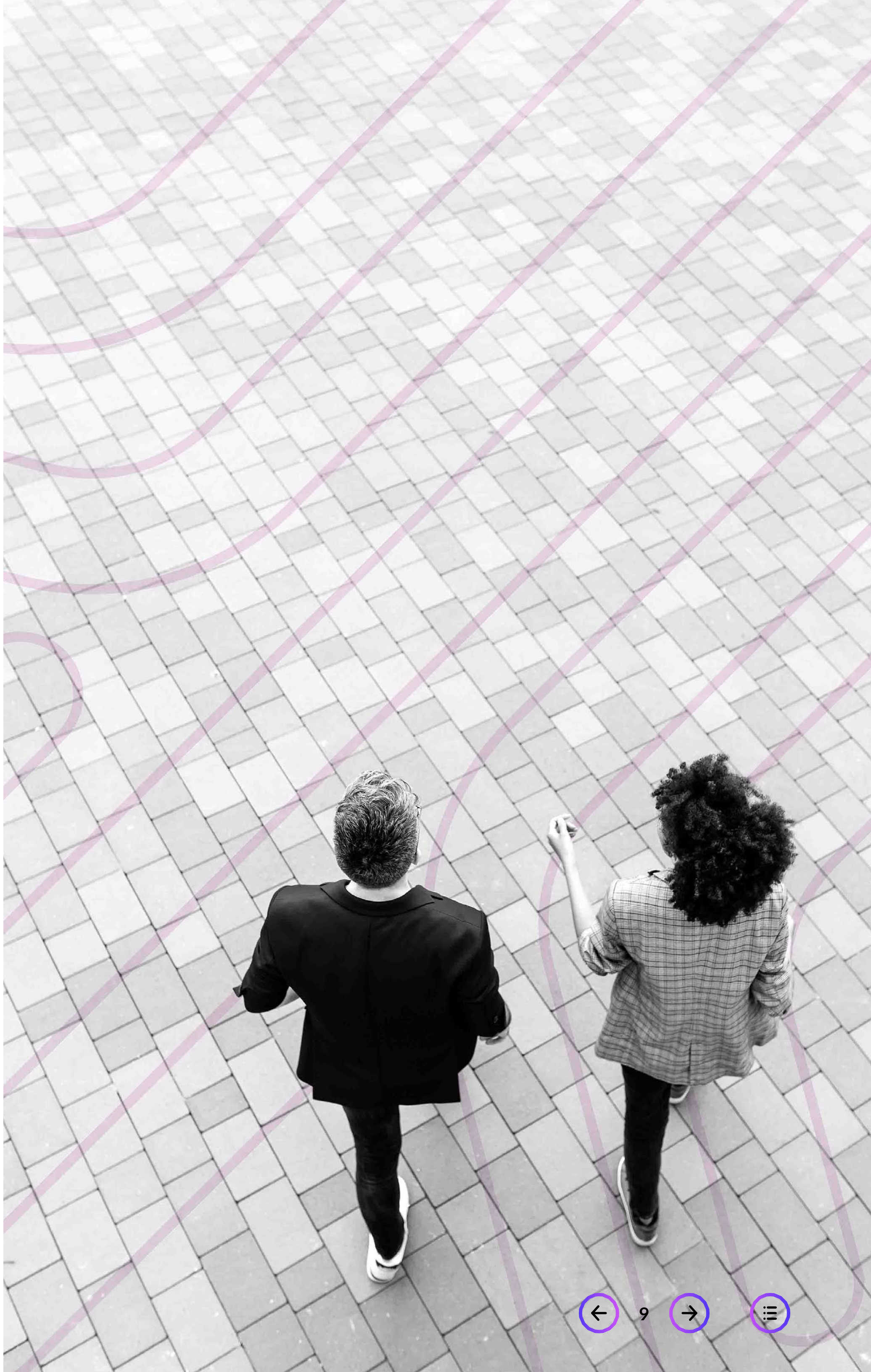
Make ILA a continuous part of the everyday operational fabric every time new processes, technologies, or trends within the business, or the larger market, warrant a review and update of the IBM i application workflows. This can help drive business resiliency through productivity improvements and higher ROI on your IBM i systems.

Get started on your modernization journey

Refactoring a monolith to microservices can help organizations gain agility and new tech capabilities to meet growing demands. However, the adoption of a microservices architecture comes with its own set of challenges. Every company's modernization path is different so identifying the best strategy that delivers the most ROI can be daunting.

Tools like Rocket® Intelligent Legacy Modernization help companies navigate their automation and modernization journey effortlessly by providing insights on where to start, how to implement, and what to transform.

The race to the cloud and microservices is on; get in touch with us to know how you can steadily sprint to the finish line.



About Rocket Software

Rocket Software is a 32+ year partner of IBM i businesses looking to innovate on the platform through industry-leading modernization and automation approaches. We have a comprehensive portfolio of modernization and automation tools, including:

- Rocket® Modern Experience enables you to easily build modern user experiences for your mainframe and iSeries applications without relying on developers with COBOL or RPG programming expertise. It also enhances your employees' and customers' web and mobile experiences while increasing productivity.

- Rocket® Process Insights, a visual tool that lets you see your workflows in totality, giving you the information to build a smart IBM i modernization plan. It tracks how your organization engages with the data and business logic of your IBM i applications and helps you build a data-driven strategy that eliminates redundancy and wasted time within your workflows. Then, you can leverage Rocket modernization solutions or the solutions of your choice to build modern user experiences and workflows that provide real results to the business.

- Rocket® Process Automation, the only RPA solution to deliver quick and significant ROI from automating your IBM i processes. It removes tedious, manual work that causes bottlenecks, introduces errors, and limits innovation without needing in-house legacy development expertise.

- Rocket® Process Integration enables businesses running legacy/IBM i applications to build workflows and innovative experiences that align with how customers and employees engage with your business and not how IT is built.

The future won't wait—modernize today.

Visit RocketSoftware.com >

Book a demo

Rocket software