

HALF OF YOUR APPLICATIONS NEED MODERNIZING BY THE END OF 2022. WHAT'S YOUR PLAN?

I was recently reading Stuart Stent's CIO.com article, "[The Crucial Role of Application Management in a Cloud Operating Model](#)," in which he notes that "IDC predicts that to gain business agility, enterprises will need to modernize over half of their existing applications by 2022, using turnkey cloud-native development and deployment services."

Organizations are looking for all the advantages of moving both their software development teams and their software products to cloud-based platforms, including:

- Lower-cost, hosted offerings for customers
- Improved application performance and customer satisfaction
- Enhanced application agility, scalability, and maintainability
- Secure, developer-ready cloud environments to accelerate innovation
- Increased developer productivity through the use of cloud-native architectures and services

Applications that are targeted for movement often require modernization to fully leverage how the cloud can help achieve the advantages listed here. Modern application architectures employ microservices, serverless computing, containerization, multi-tenancy, cloud-native services, and other cloud features. Many applications require refactoring, often with significant architectural changes.

Modernizing an application to take advantage of the appropriate cloud-native architectures and services can be complex, and requires expertise in cloud technologies, processes, and tools. Each application requires a detailed assessment and a determination of the appropriate refactoring to meet business needs. Decisions need to be made regarding critical refactoring vs. enhancements that can be included later. Unless an organization has already performed multiple cloud application modernizations, there may be a steep learning curve and possible missteps before you achieve success. Often organizations don't know where to begin.

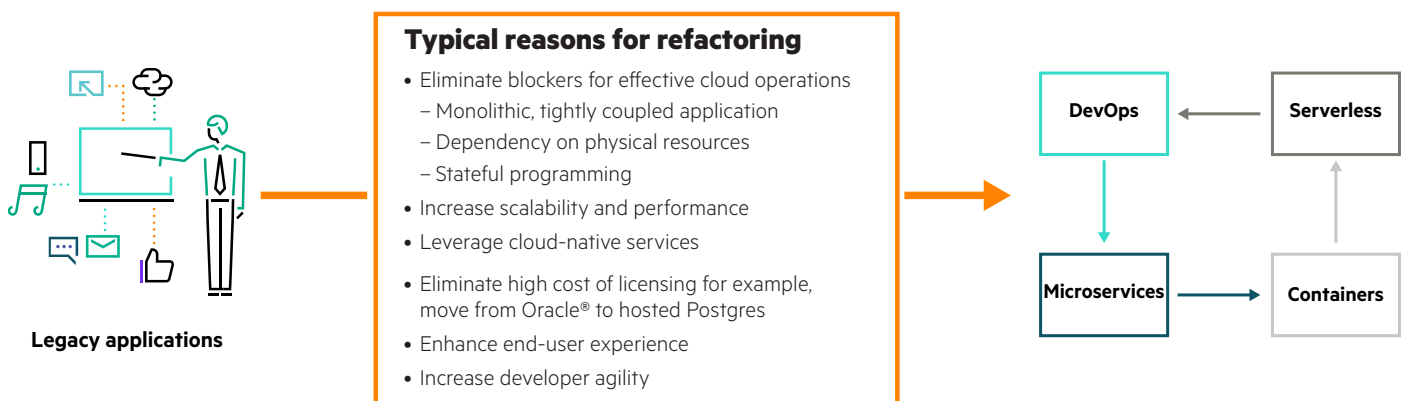


FIGURE 1. Why refactor applications?

PLANNING FOR SUCCESS: A 7-STEP BLUEPRINT

1. Assess application cloud-readiness and design the cloud architecture to support modernization

Both manual and automated code analysis can be used to assess an application's cloud readiness. Key determining factors include hard-coded connections, remote method invocations, and thread management. In addition, review the application architecture to assess the applicability of decomposing business logic into cloud frameworks, such as microservices and serverless, and the suitability of incorporating cloud-native services to meet business, performance, and scalability needs. A thorough evaluation of non-functional requirements is necessary for driving the design of appropriate [cloud infrastructure](#) to support the modernized application. When designing the infrastructure, consider best practices such as common shared services for logging, monitoring, and security. The infrastructure design should also include the frameworks and tools your development team needs to accelerate innovation and facilitate agile processes.

2. Re-imagine the user experience in ways that are customer-centric and market-relevant

While most organizations understand the technical benefits of refactoring their application codebase, many don't consider one of the key factors driving customer loyalty—the user experience. Experience design techniques ([expanded in this article by M Safder](#)), or empathy workshops uncover what customers really want. So, design sessions can quickly turn ideas into interactive prototypes and drive the most creative solutions. These workshops and design sessions should include real customers who can provide iterative feedback as prototypes are developed and refined. Using these techniques ensures that the modernization effort includes not only modernizing the technology but also the customer experience. Take this opportunity to delight your customers. If you don't, your competition will.

3. Quickly validate technical decisions and end-to-end functionality using a proof of concept (POC)

An organization's stakeholders often want a high degree of comfort that their software product will perform as expected in the cloud. There may also be technical decisions that need to be evaluated and validated. A POC is a cost-effective way to quickly demonstrate end-to-end functionality to stakeholders, investors, and even customers; validate emerging technologies; and/or prove value. A POC is also a good way to create and validate the foundational infrastructure and technical components needed for the refactoring sprints. Time to market is critical. Build out only the minimal cloud infrastructure needed to support the POC and develop only the POC elements that are critical for the decision to move forward.

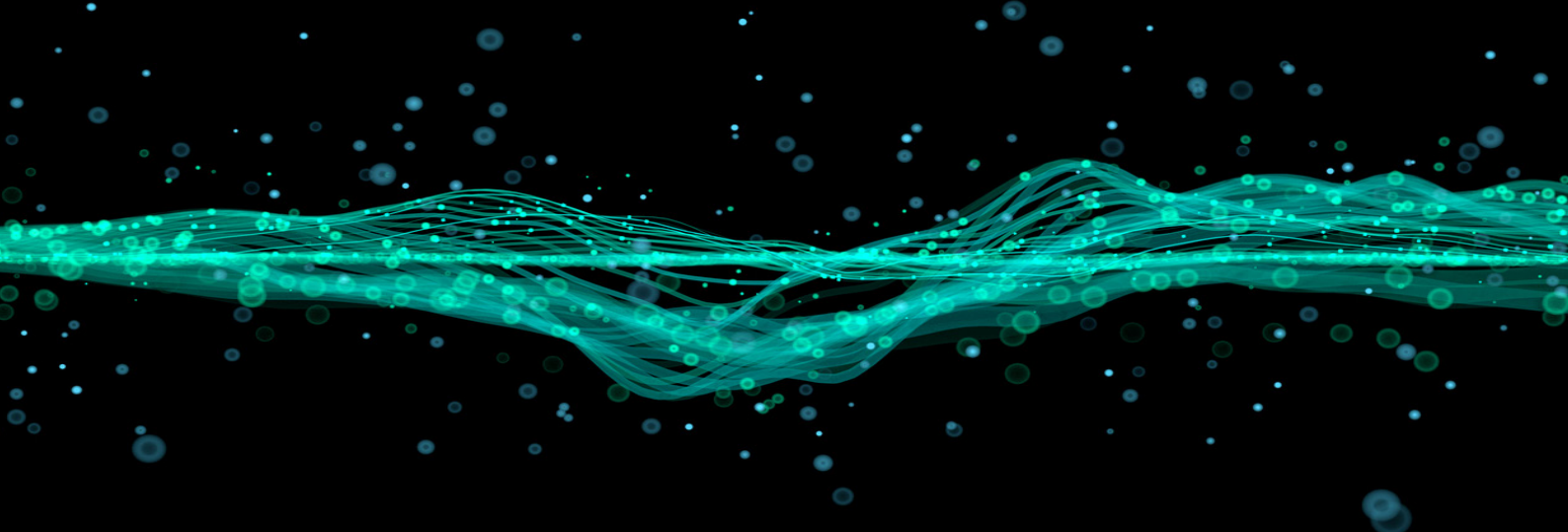
4. Define the minimum viable product (MVP) and development road map

Modernization requirements, driven by the application assessment results, architecture design, and experience design, should be translated into high-level user stories that will be decomposed into project backlog during the refactoring sprints. The user stories can then be prioritized to define the MVP to accelerate the time to market. Keep in mind that not all modernization requirements need to be implemented during the initial refactoring. Choose those requirements for the MVP that will provide the best return on investment and will address the most critical business needs. The goal of defining MVP is to get the cloud-based software product into the hands of customers as quickly as possible. The remaining user stories become part of a phased plan that serves as a road map for achieving longer-term business goals.

ACCELERATE WHAT'S NEXT WITH EXPERT ADVICE AND IMPLEMENTATION

Digital Next Advisors and **Advisory and Professional Services** experts from HPE Pointnext Services map technology and offer best practices advice to help you deliver on digital ambition. To find out more about Digital Next Advisory or engaging with an HPE Digital Advisor contact us at: digitaladvisor@hpe.com.





ABOUT DEBBIE MOUNTS

Debbie Mounts has over 30 years of experience in software development, cloud solutions and services, and product management. She currently leads the Content Strategy and Enablement Team in the Advisory and Professional Services organization for HPE Pointnext Services. She is responsible for developing strategic, thought-leading content, and learning programs that empower HPE to meaningfully engage with clients.

Make the right purchase decision.
Contact our presales specialists.



Chat now (sales)



Call now



HPE support



Get updates

 **Hewlett Packard
Enterprise**

5. Build out new or optimize existing cloud infrastructure to support modernized applications

A new cloud environment will need to be built or an existing cloud environment modified, as designed during the architecture assessment, to support the given, modernized application. Wherever possible, infrastructure as code should be developed to define the infrastructure. This will enable automated, repeatable ease of provisioning. Start with the development environment so that the refactoring sprints can begin and then build out test, user acceptance testing (UAT), and production environments as development progresses. To speed development cycles, deploy a continuous integration (CI)/continuous delivery (CD) toolchain that includes the integration of automated testing and code promotion, along with any cloud frameworks and tools the development team may need.

6. Modernize the application using an agile, MVP approach

Once the MVP is defined and the development environment is built, refactoring can begin. Use an agile, sprint-based approach that allows rapid response to changes in requirements and priorities, facilitates CD of working features to stakeholders for feedback and acceptance, and drives time to market. Continually validate the user experience with real customers to ensure the end product will keep them engaged. Optimize the infrastructure, through updates to the infrastructure as code and maintain the CI/CD scripting and automated testing pipeline throughout the development lifecycle.

7. Make automated testing an integral part of your modernization strategy

Move testing as far left in the process as possible. Leverage automated testing in your CI/CD toolchain to maintain feature parity before and after the modernization process. Testing should include unit, functional, destructive, and load testing. Testing should include applications, configurations, and infrastructure, providing confidence as you work through the release process.

ARE YOU READY?

There is tremendous value wrapped up in organizations' existing applications; they power the business. Modernizing those applications while maintaining that value increases business agility and improves application scalability, resiliency, and usability. By following this modernization blueprint, organizations can get started on a successful modernization initiative and see results against their business goals as quickly as possible. Following this blueprint will provide the significant business value that's motivating you to undertake wholesale application modernization.

For further information, visit hpe.com/greenlake/cloud-adoption-framework.

LEARN MORE AT

hpe.com/digitaltransformation

© Copyright 2022 Hewlett Packard Enterprise Development LP. The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Oracle is a registered trademark of Oracle and/or its affiliates. All third-party marks are property of their respective owners.
a50005904ENW