

## HOW ZERO-DISRUPTION MODERNIZATION WORKS

Selecting and designing the correct modernization strategy can be a daunting task for companies. Fear of disrupting business continuity is a common reason for delaying modernization initiatives. This seven-layer approach can minimize disruptions.



Businesses across industries constantly strive to be relevant to their clients, resilient to market shocks, and responsive to market forces. However, they often find that their aging software and technology infrastructure hinders these efforts. The simple answer is to modernize — but modernizing is not simple, easy, or cheap. Not only that, it can disrupt business continuity and stakeholder experiences.

Not all modernization strategies are created equal. The risk of business disruption during the modernization phase can be greatly reduced, even to near zero, through carefully selecting a modernization strategy that aims for relevance, resilience, and responsiveness.

## Selecting the right modernization strategy

The right strategy to modernize an enterprise IT estate depends on the organizational context, risk appetite, and modernization goals. We address complex modernization projects in this article — projects that will enable incumbent enterprises to compete

with digital natives. For these complex projects, there are three modernization strategies: in legacy, around legacy, and off legacy. (See Figure 1.) These strategies fall on a continuum from superficial to structural changes and, thus, range from less invasive to more invasive.

*In-legacy efforts* keep much of the existing technology infrastructure and concentrate on creating efficiencies and agility in existing applications. This acceleration approach is least invasive and is typically used with systems of record.

*Around-legacy modernization* seeks to enable digital readiness by speeding up the deployment of APIs, and offloading data. This renewal approach works best with systems of record and systems of differentiation. It is also considered a less invasive approach.

*Off-legacy modernization* is a transformational effort that moves systems to the next-generation architectures. It's more invasive but can future-proof the system architecture with no loss of functionality. This works for the same systems as the others but also with systems of engagement.

Companies often struggle to balance the trade-offs in deciding which strategy to take. The approaches can range from narrow, cosmetic changes (non-invasive) to broad, deep changes (invasive).

Less invasive strategies are those that impact the organization in pockets — for instance, some staff, some applications, and some processes. They are typically less expensive, modernize minimally, and for the most part are only incremental changes. They include changes such as rehosting, refactoring, and integration. These less invasive strategies are also less likely to have a significant impact on the end-customer experience. Non-invasive strategies often form the basis for other modernization initiatives down the road.

More invasive strategies are those that impact multiple stakeholders and multiple layers of application, and have a significant impact on processes. They are typically more expensive, wholesale IT changes that involve package implementation and/or reengineering.

Figure 1. Modernization strategies are tailored to the system being modernized

Legacy effort	Modernization strategy (ART)	Benefits	Typical workloads
<i>In</i>	<b>Accelerate (A)</b> agility and cost efficiency	<ul style="list-style-type: none"> <li>• Performance improved</li> <li>• Application maintenance time reduced</li> <li>• Future modernization efforts reduced</li> <li>• System's flexibility and scalability improved</li> <li>• Reduced time to market</li> </ul>	<ul style="list-style-type: none"> <li>• Systems of record</li> </ul>
<i>Around</i>	<b>Renew (R)</b> to enable digital readiness	<ul style="list-style-type: none"> <li>• APIs deployed quicker</li> <li>• Overall system more flexible and agile</li> <li>• System efficiency improved and duplicated business functions are mitigated</li> <li>• Web and mobile platform supported with very minimal changes to existing system</li> </ul>	<ul style="list-style-type: none"> <li>• Systems of record</li> <li>• Systems of differentiation</li> </ul>
<i>Off</i>	<b>Transform (T)</b> to next-generation architecture	<ul style="list-style-type: none"> <li>• Maintenance and extensibility improved exponentially</li> <li>• Improved legacy rule extraction and logic generation</li> <li>• Improved business agility</li> <li>• Future-proof architecture</li> </ul>	<ul style="list-style-type: none"> <li>• Systems of record</li> <li>• Systems of differentiation</li> <li>• Systems of engagement</li> </ul>

Source: Infosys



These efforts thoroughly modernize IT domains but take longer, and can be more disruptive to the business operation if not planned properly. Invasive efforts, however, deliver the maximum benefit to the organization and the maximum value to the end consumers.

## Non-invasive strategies

Non-invasive strategies typically offer surface-level changes to a few components of the IT domain. These are normally used when a company does not want the big operational risk of replacing all legacy systems, or the company can afford to only change critical functions. These strategies are more suited to situations where only a small part of the organization supports modernization, the IT architecture is acceptable, the complexity of the system is manageable, current interfaces are adequate, and no major changes are needed in the integration logic.

Examples of non-invasive modernization include those that

accelerate and renew legacy systems (see first two rows of Figure 1), such as

- Migrating technology frameworks to cloud
- Migrating to open source application servers
- Migrating functions to container platforms
- Rehosting mainframes
- Externalizing business rules

Below are three broad categories of non-invasive strategies (rehosting, refactoring, integration) and when they should be implemented.

### Rehosting

With rehosting, companies typically move workloads from legacy hardware to UNIX-like operating systems with emulators running on commodity hardware. This requires very little, if any, recoding as the entire application is just switching locations. This is typically done to reduce the mainframe's computing load — and subsequent costs — and increases

the application's availability to other company apps.

### Refactoring

Refactoring changes the code of an application to make it run more efficiently or to improve technical quality. This practice usually involves removal of redundant and duplicate code and the implementation of common libraries. Removing duplicate code means that new changes are needed only in one place, which significantly reduces development time. Common libraries can reduce the memory footprint of an application because functions can be shared by many applications.

### Integration

In the non-invasive process, a company will integrate with a new system or a third-party vendor to gain information not contained in legacy systems. A simple version would be creating a link between the company application and one from a customer information data service.

## Invasive strategies

These deeper changes generally require companies to rework most of their core IT domains. The strategies are normally used when the benefits outweigh the operational risks of replacing all legacy systems or there are no significant budget constraints. In this case, senior management is generally willing to lead the entire change project.

Invasive strategies are more suited to situations where the current "spaghetti" IT architecture is stunting business growth or the company needs a new, flexible, but standardized architecture for application integration. Below, we outline the major forms of invasive strategies and how or when they should typically be used.

Examples of invasive modernization are those that transform legacy IT (see first two rows of Figure 1), such as

- Re-architecting legacy applications to cloud native infrastructure

- Migrating traditional relational database management systems to NoSQL-type systems such as MongoDB
- Simplifying every stage of the app development and delivery process by modernizing with low-code/no-code platforms
- Simplifying business processes and business process management

Broadly speaking, there are two kinds of approaches: Package implementation and reengineering from the ground up.

### Package implementation

Package implementation replaces current software with the latest commercial packages available from SAP, Oracle, IBM, Microsoft, and similar vendors. For example, companies can use a homegrown human resources system, or they can implement a system package from a provider such as Oracle. These packages, however, still require some customization based on the company's processes. The best

practices are to change company processes rather than customizing the package. Using pre-built packages also allows for easier maintenance and security upgrades.

### Reengineering

Reengineering is the most invasive strategy since it usually involves business process changes besides the change to the entire technology landscape and also has an impact on the way IT operations are managed. Target architecture quite often comprises new-age open source and cloud technologies. Also, the change includes modernizing data, implementing niche technology products, and customizing for specific functionalities. When moving to a new platform, a company typically switches to cloud services, uses open source software and libraries, and converts current applications to containers and API-enabled microservices.

A summary of these strategies is depicted in Figure 2.

Figure 2. Reasons to choose non-invasive or invasive modernization strategies



Source: Infosys

## Resolving the disruption issue

Although non-invasive strategies typically carry a low risk for disruption, they also tend to generate far lower customer value than invasive strategies do (see Figure 3). Even so, that decision is often a suitable compromise that balances risk versus reward, and cost versus savings, for many companies.

Until recently, a large modernization project would be conducted separately from the legacy system. Eventually, there would be a sudden switch to the new system. But that was rarely the end of the process. Often, issues were discovered that prevented the new system from being deployed and the switchover date would be postponed.

Each time, the company was at risk of losing business until the new system was finally deployed.

Now, companies can develop a modernized system in coexistence with the legacy system so that IT infrastructure and applications are gradually transitioned.

## How to achieve zero-disruption modernization

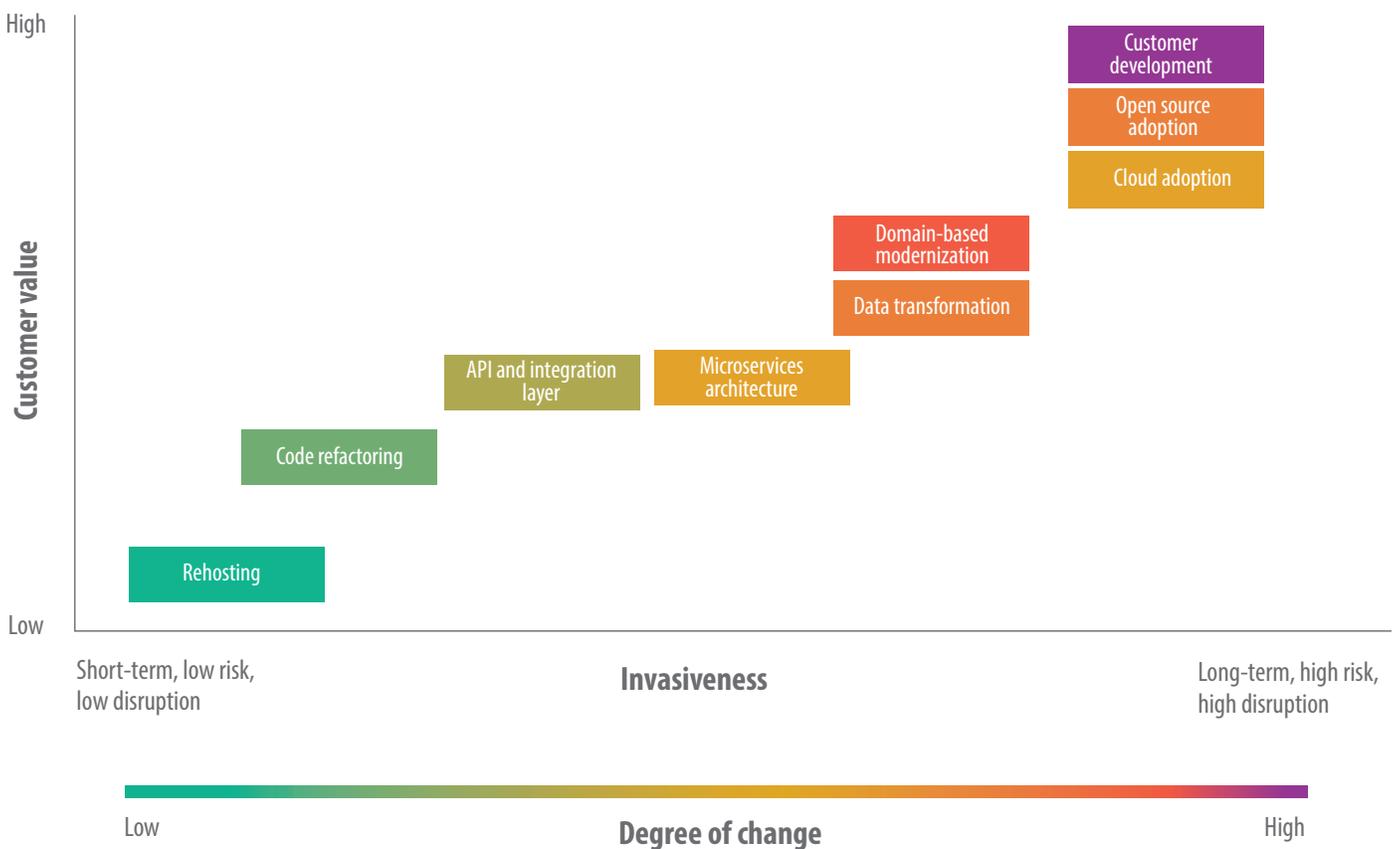
Understanding what modernization strategies exist and which are likely to be more invasive and therefore more disruptive is useful. But how do companies actually achieve zero-disruption modernization?

Design principles, methodology, and technologies form the three main building blocks of zero-disruption modernization

Zero-disruption modernization features three main building blocks: design principles, methodology, and technologies.

When designing a modernization project, it is important to put the company's clients first and ensure changes are introduced incrementally, without a sudden and abrupt disruption. When the end consumer is an enterprise, that enterprise's systems should see minimal to zero changes to consume the services.

Figure 3. Each option for modernizing has risks and rewards in the short and long term



Source: Infosys

Business operations need to seamlessly transition from supporting the legacy estate to the modernized model. One key to this seamless transition is training existing staff on technical skills, in addition to new development and operation practices.

Using an Agile-based delivery method minimizes disruptions because of the incremental rollout. This method includes, for instance, iterating on minimum viable products for quick releases, feedback, and continuous improvement. The planning phase should specifically consider the customer's business priorities, business functions, feature dependencies, complexity of client's business, geography, and so on.

In addition to putting clients first and using Agile methodology, it is also critical to explore all the ways that the legacy platform can continue to be maintained and used. This will help keep the cost of modernizing lower and less disruptive. Quality

assurance planning will also ensure that the modernized estate is optimally functional.

It is crucial that organizations adopt an architecture-first approach to technology choices. In contrast, a product-first approach (which builds on a pre-decided product) results in a suboptimal solution, with a prolonged effect on the cost profile of the modernized estate. For example, cloud-agnostic programming creates future flexibility by not relying on a cloud vendor's specific technology. Similarly, open architecture software also allows companies to implement future changes more easily.

Finally, many companies overlook the people and talent side of modernization. The workforce needs to be upskilled and reskilled not only on the new technologies, but also at the new ways of working the new technologies enable. A carefully crafted staff skilling program should be an integral part of the

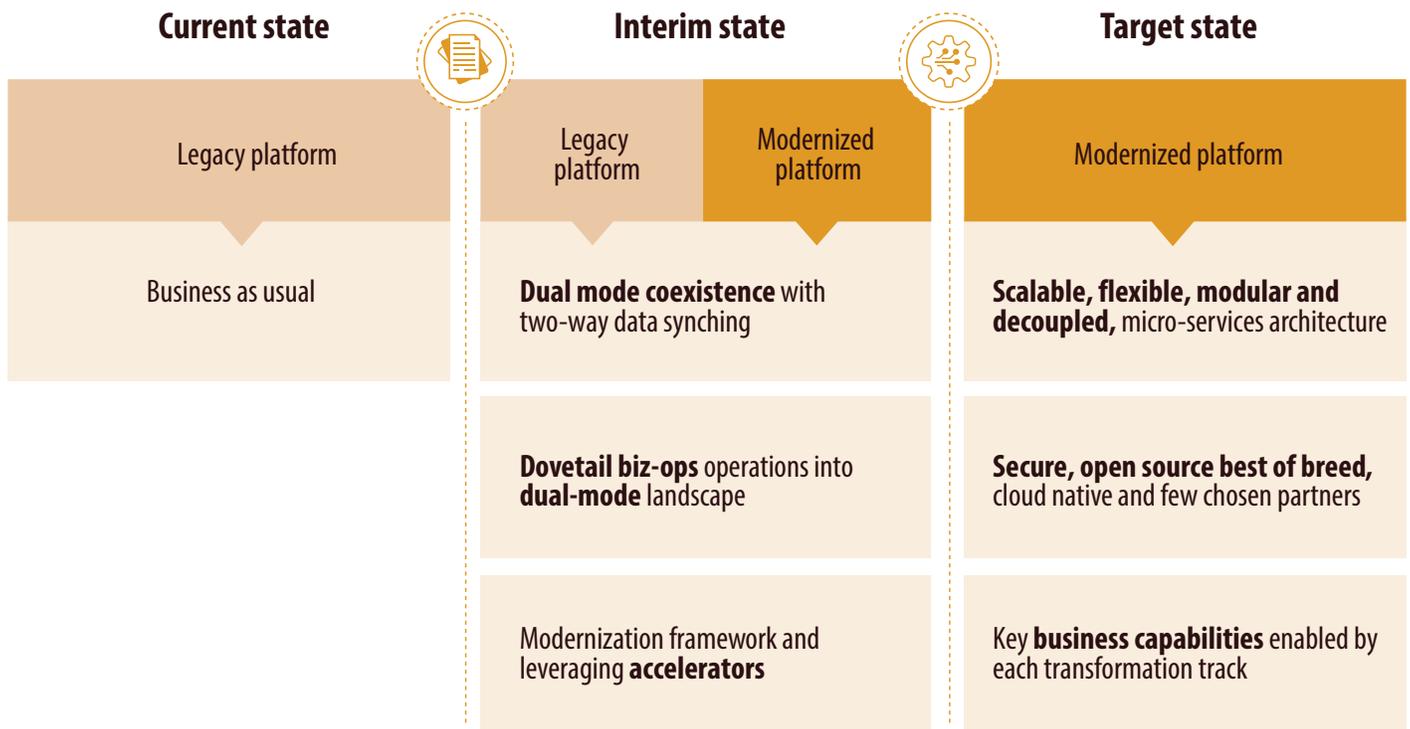
modernization program plan and can minimize disruptions.

## Phases

The zero-disruption modernization model typically follows a three-phase approach: current (legacy phase), interim (coexistence phase), and target (modernized phase). Under this approach, small aspects of the legacy system are changed incrementally while the system is still in use (see Figure 4). This model is flexible and can be tailored to each enterprise.

The most crucial time is in the interim phase where both legacy systems and modernized components coexist. This is the time when disruptions could occur and careful planning from the previous phase becomes most important. Monolithic applications are incrementally converted to microservices. And an operational data repository acts as a bridge between modern micro-databases and legacy databases.

Figure 4. The three phases of zero-disruption modernization



Source: Infosys

## Seven layers of zero-disruption modernization

There are seven layers that — when managed correctly — help execute a modernization project that has zero disruptions (see Figure 5).

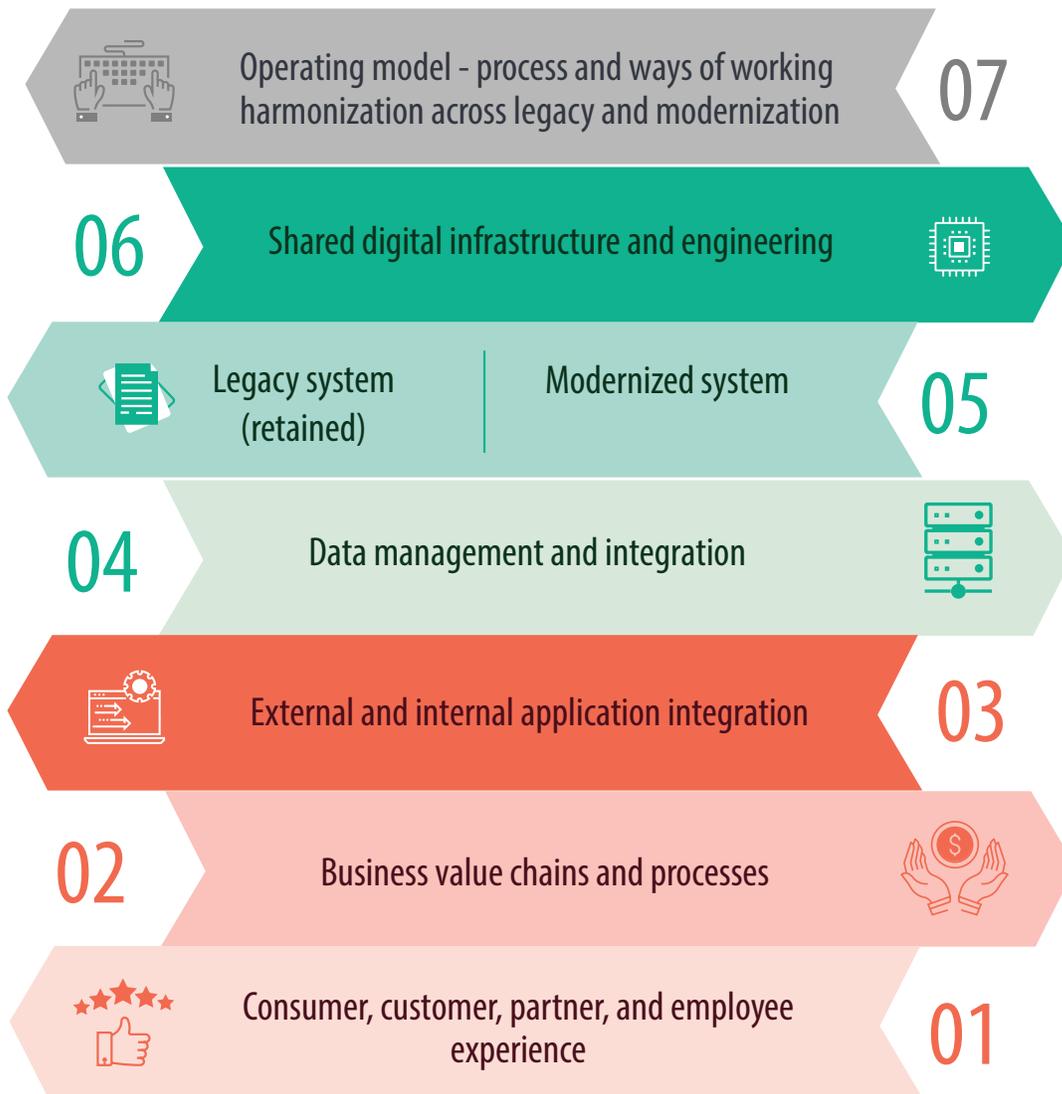
Throughout the process, microchange management should be applied at every layer to ensure a consistent and stable end-to-end user journey and flow. Microchange management is important so companies can realize

early value for customers without any disruption to business operations. It also breaks down change into small, orchestrated steps,<sup>1</sup> driving adoption of the new experience, harmonized business processes, and new ways of working, among others. These microchange interventions can be carried out in agile sprints, so change occurs one micro step at a time. As changes are implemented, outcomes need to be frequently measured to see if there is any “course correction” needed through iteration.<sup>2</sup> In the terminology of our approach: Micro is the new mega.<sup>3</sup>

### Layer 1: Consumer, customer, partner, and employee experience

In addition to microchange management, well-planned user experience integration strategies help ensure continuity during the coexistence phase. Early on, companies should consider all the relevant stakeholders and the experience they will have while the modernization is ongoing. As mentioned above, employees should be skilled and upskilled as part of the stakeholder considerations. During modernization,

Figure 5. Nondisruptive modernization requires focus and careful planning across several areas



Source: Infosys

companies can get feedback from external clients on products and services, operations, and go-to-market teams. From there, an internal, cross-functional team consisting of, for instance, product managers, customer experience specialists, research and development, IT transformation teams, and operations, can continually iterate to meet user needs and provide exceptional experiences.

## Layer 2: Business value chains and processes

Focusing on business value chains and processes will also drive maximum value and minimize risks during the coexistence phases of legacy and modernized systems. Specifically, companies should prioritize the delivery of changes based on value stream mapping. They can employ user journey and business

process impact planning during the coexistence and modernized phases.

In this step, we recommend first looking at organizational factors such as business value chains and the to-be process definition. These factors can be considered, along with the business case, to select the best modernization strategy. Once that is done, a pilot program can be implemented, using a few medium-risk, high-impact apps — ideally by leveraging a partner's expertise.

## Layer 3: External and internal application integration

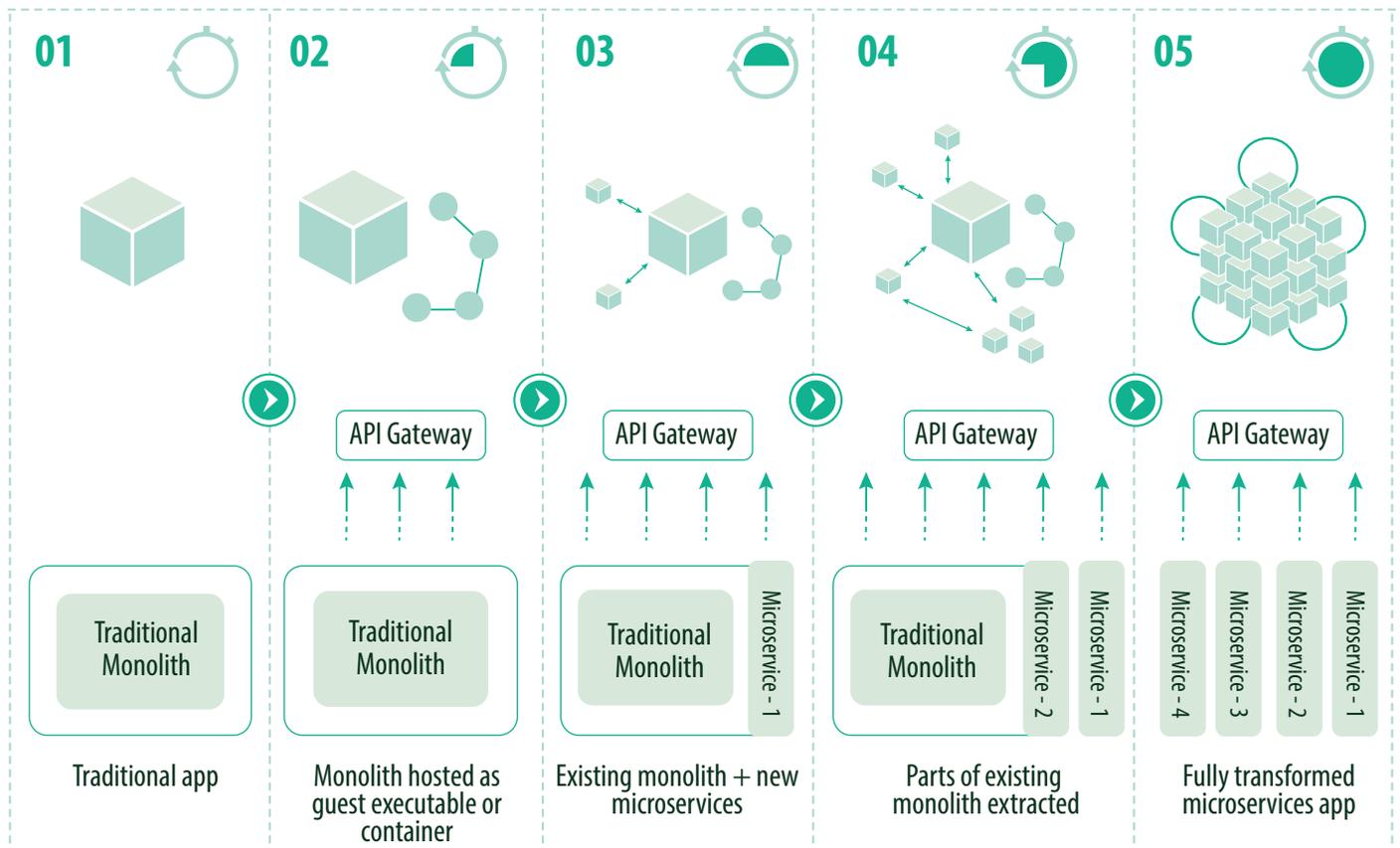
When a technology estate is modernized, there will invariably be changes to the interfacing systems — both internal enterprise systems and systems belonging to client organizations. These can be

batch-based systems exchanging data files or online transaction processing systems.

The key focus of this layer is to architecturally ensure zero or minimum changes are required on the interfacing estates due to any modernization-related changes on the system undergoing modernization. A key step is to comprehensively identify all upstream and downstream interfaces. Any data format or other changes should be handled by the system being modernized through appropriate data enrichment mechanisms and data quality checks.

This layer also ensures the incremental change of the interface to the external world through a carefully crafted migration from a monolith to a microservices-based organization (see Figure 6).

Figure 6. In a zero-disruption approach, modernization is incremental



Source: Infosys



## Layer 4: Data management and integration

Coexistence of the legacy and modernized systems is key to modernizing with minimal to zero disruption. For optimal coexistence, the right data management and integration strategy is critical. Companies should thus give a lot of careful thought to the data management strategy.

Coexistence of the legacy and modernized systems is vital to modernization with minimal to zero disruption

One way to manage data is to create a transitional operational data store on the cloud. The main reason for this store is to keep data from the legacy and modernized application estates in sync. This ensures no loss of service, regardless of whether a customer gets

routed to the legacy system or the modernized system.

We recommend storing data in this way through five key steps:

1. Establish an operational data repository (ODR) on the cloud
2. Migrate data all at one time to the ODR; add a regular forward sync of the transaction data from the legacy database to the ODR
3. Populate the online transaction processing (OLTP) system micro-databases with the relevant data
4. Regularly sync the transaction data captured through the OLTP micro-databases into the ODR
5. Reverse sync the transaction data captured on the modernized OLTP micro-database into the legacy database

Figure 7 shows this phase in greater detail. The end result is that all users — customers, operations, IT, helpdesk, back office, and partners —

have a seamless experience, without disruption.

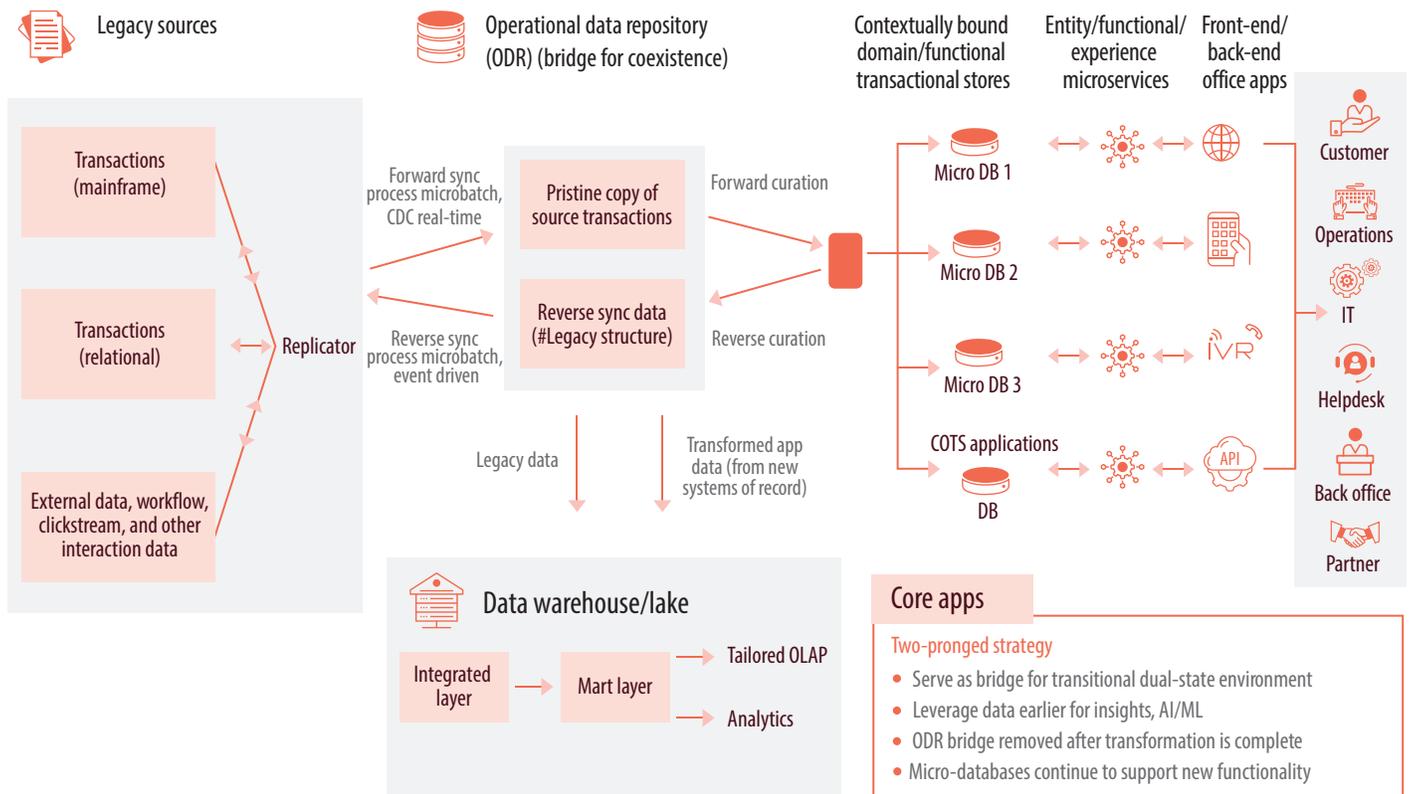
When the move to modernized systems is complete, the bridge (i.e., the ODR) can be decommissioned or repurposed for other requirements, depending on the context. The modernized system will continue to operate through the OLTP systems, backed by micro-databases.

## Layer 5: Coexistence of the legacy and modernized systems

Having established the interface plan in Layer 3 and the ODR in Layer 4, the foundation to operate the legacy and modernized systems in parallel is in place. Layer 5 devises strategies and tech approaches to route the traffic to the appropriate system (legacy or modernized). The route is based on the client and functionality migration plan arrived at during Layer 2.

Key at this stage for coexistence of legacy and modernized systems is an

Figure 7. Taking a data-first approach will unlock data on modernized platforms



Source: Infosys

efficient abstraction layer strategy. This layer straddles both modernized and legacy systems, and legacy mid-tiers support this abstraction layer. This layer interacts with both systems. Specifically, it interprets the nature of the requests from user interfaces and process layers. It routes the request to the right system, depending on the client, functionalities, and data that the underlying systems support.

For the back office, API gateways must use the right routing logic so that applications can coexist. This API gateway and logic will ensure that there is a single internal landing page, seamlessly performing actions on the legacy or modernized platform.

When applications coexist in this well-designed way, customers can focus on delivering modernized capabilities to their end users, without risks. This solution allows for a tapered/throttled approach to onboard clients in an incremental manner depending on criteria such as size, complexity, importance of customer, etc.

### Layer 6: Shared digital infrastructure and engineering

This layer focuses on offering a strong and secure digital infrastructure to all users to complete the modernization story. Specifically, this layer enables business and technology agility to power the user experiences that drive growth and innovation.

We recommend an integrated, platform-based approach that creates a shared digital infrastructure. The two essential elements for this phase are cloud and partnerships.

Cloud is essential to this approach, because it can harness the power of DevSecOps, operational resilience and migration and modernization tools, and test automation to provide an enterprise-level view of all interventions.

Specifically, adopting DevSecOps automation and capabilities

through a platform approach to cloud significantly accelerates the transformation journey. Integrating capabilities of infrastructure provisioning, development, test automation, security testing, and deployment tools into a single DevSecOps platform on cloud offers accelerated and secure releases. Self-service virtual desktop provisioning and one-click development and test environment provisioning on cloud allows rapid, standardized, and secure user onboarding. It also reduces wait times and ensures teams hit the ground running — faster, more productively, and earlier.

Because of the scale and complexity of modernization, companies should rely on a rich partner ecosystem. Ideally, this ecosystem will include startups, hyper-scalers, private cloud companies, and an array of partners that specialize in relevant products, tools, apps, platforms, security, data center, and infrastructure solutions and services.

This combination of cloud and ecosystem partners will deliver a solution that leverages a suite of tools across all layers — users, services, security, support, business, and so on. This integrated approach to modernization is secure by design.

### Layer 7: Operating model

An essential layer in the modernization process is an operating model that encourages interactions between business operations and technology services, the transformation teams and innovation groups, and the business-as-usual application development team. This will help ensure processes and ways of working are harmonized across legacy and modernized systems as well as across teams. This is essential since other teams may be working on projects that will directly impact the work of the modernization team and vice versa. Such a harmonization will ensure synergies that will accelerate the modernization exercise.

A focused initiative to identify, harmonize, and scale processes and operations can help legacy and modernized systems work better together

Legacy and modernized systems will work better together through a focused initiative to identify, harmonize, and scale processes and ways of working. We recommend that companies identify key client and supplier stakeholders and also analyze tool-based process extracts and documents. They should then follow a systems or design thinking process,<sup>4</sup> focusing on the end-user experience. For instance, they can conduct workshops and interview stakeholders to analyze the current state and to identify gaps and improvement areas.

The inputs from the above activities should be used to gradually synchronize ways of working across the legacy and modernized systems. We also recommend training employees to increase adoption of new ways of working. A gradual, microchange-driven approach will ensure that there are no disruptions while this is implemented. This method of harmonizing the operating model will lead to sustained innovation and agility across the entire organization.

Businesses often encounter a tricky situation as they modernize: It was risky to stand still, it was risky to make minimal changes, and it was risky to move forward. That sometimes resulted in paralysis. However, companies no longer need to choose among just these options. They can find a path that takes the best of all these strategies and leads to zero disruptions.

## References

1. [Break down change management into small steps](#), Jeff Kavanaugh and Rafee Tarafdar, May 3, 2021, Harvard Business Review.
2. [Break down change management](#).
3. [The live enterprise: Create a continuously evolving and learning organization](#), Jeff Kavanaugh, Rafee Tarafdar, Jan. 26, 2021, Infosys.
4. [Systems design: Design thinking 2.0](#), Harry Keir Hughes, Nur Karadeniz, and Samad Masood, January 2020, Infosys Insights.

## Contributors

### **Sundaram Vishwanathan**

*Delivery Manager, Application Development and Maintenance, Infosys*

### **Vageesh Patwardhan**

*Senior Principal, IP Deployment and Commercialization, Infosys*

## Authors

### **Arvind Balasubramanian**

*Global Delivery Partner, Financial Services, Infosys*

### **Rafee Tarafdar**

*Senior Vice President and Unit Technology Officer, Strategic Technology Group, Infosys*

### **Naresh Duddu**

*Associate Vice President and Senior Delivery Manager, Head - Cloud & Open Source, Modernization Practice, Infosys*

### **Isaac LaBauve**

*Infosys Knowledge Institute*

---

## About Infosys Knowledge Institute

The Infosys Knowledge Institute helps industry leaders develop a deeper understanding of business and technology trends through compelling thought leadership. Our researchers and subject matter experts provide a fact base that aids decision making on critical business and technology issues.

To view our research, visit Infosys Knowledge Institute at [infosys.com/IKI](https://infosys.com/IKI)

---

For more information, contact [askus@infosys.com](mailto:askus@infosys.com)



---

© 2021 Infosys Limited, Bengaluru, India. All Rights Reserved. Infosys believes the information in this document is accurate as of its publication date; such information is subject to change without notice. Infosys acknowledges the proprietary rights of other companies to the trademarks, product names and such other intellectual property rights mentioned in this document. Except as expressly permitted, neither this documentation nor any part of it may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, printing, photocopying, recording or otherwise, without the prior permission of Infosys Limited and/ or any named intellectual property rights holders under this document.